[19]中华人民共和国国家知识产权局

[51]Int. Cl[7]
G06F 15/163
G06F 13/14 G06F 17/30

# [12]发明专利申请公开说明书

[21]申请号 00123530.3

[43]公开日 2001年2月14日

[11]公开号 CN 1283827A

[22]申请日 2000.8.18 [21]申请号 00123530.3

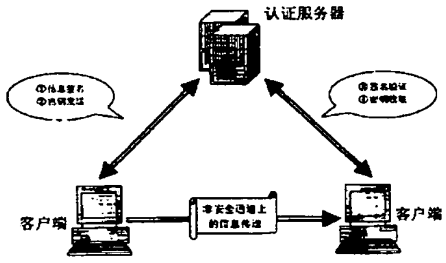
[71]申请人 郝孟一
地址 100095 北京市海淀区温泉乡环山村64楼4单元12号
共同申请人 郝晓冬
[72]发明人 郝孟一

权利要求书4页 说明书15页 附图页数14页

[54]发明名称 通用电子信息网络认证系统及方法

[57]摘要

本发明提出的方案和软件系统能够实现一系列与网络信息安全有关的保障措施。本系统所用的关键技术是客户/服务器方式的网络连接。本系统以"用户帐号+密码"为基本的身份验证方式,在此基础上,为用户创建的电子信息提供网络签名及相应的签名验证,同时可以为任意两个用户之间安全可靠的传递一个或多个128比特位的随机密钥。本系统的衍生功能还包括:电子公章;用户之间私人信息的安全共享;两个客户端之间的单向身份识别或双向身份识别。特别指出的是:本系统提供的网络签名具有永久可验证性及自动时间戳等重要特性。这些信息安全服务功能,使得电子商务、电子银行、电子政务等一切"无纸化"社会行为的安全运作成为可能。




ISSN1008-4274

知识产权出版社出版

00.08.18

# 权 利 要 求 书

1、一种产生电子信息签名的方法，该方法包括以下步骤：

一个人能实现电子信息签名的前提是在网络中的一台专用服务器（也称认证服务器）上建立一个属于自己的帐号；

签名人首先用一个散列函数计算某特定信息的数字摘要，该特定信息称为源信息；

该签名人通过网络通讯将本人的身份验证信息及上述形成的数字摘要传送到已经建立自己帐号的认证服务器，这个过程称为签名请求；

认证服务器通过上述的身份验证信息判断该签名请求是否合法，如果合法，则将传送来的数字摘要输入到签名数据库中，同时还要记录该签名的形成日期和时间，然后向客户端返回一个可唯一识别该签名记录的签名序列号；如果身份验证信息非法，则返回一个错误信息；

签名人把上述的签名序列号、自己的帐号名称以及认证服务器的域名（或者是该服务器的 IP 地址）等标识信息附加到源信息中，合并后的信息就成为签名信息。

2、一种验证电子信息签名的方法，该方法包括以下步骤：

验证者可以通过任何手段接收到权利要求 1 中所述的签名信息；

验证者提取签名信息中的标识信息；

验证者通过网络通讯将该标识信息传送到权利要求 1 中提及的同一台认证服务器，这个过程称为签名验证请求；

认证服务器通过识别帐号名称以及签名序列号可以从数据库中查询到相应的数字摘要、以及相关到签名日期、签名时间等公证信息，然后将这些信息返回到客户端；

验证者提取签名信息中的源信息，并用权利要求 1 中提及的相同散列函数重新计算该信息的数字摘要；

如果计算得到的数字摘要和服务器返回的数字摘要完全相等，则表明该签名信息是完全合法的，而且在传送过程中没有被篡改过；与此同时，验证者还获得了该签名形成的公证日期和时间。

3、如权利要求 1 所述的方法，其特征在于签名请求中的身份验证信息要求填充 4 项内容，各项内容的形成步骤如下：

签名人将自己的帐号名称以明文方式填充到身份验证信息中，为内容 1；

签名人将密码原文进行散列函数运算，形成 128 比特位的密码摘要，该密码摘要将不会在网络中进行传送，只在本机使用；

将上述的密码摘要和帐号名称再合并进行散列函数运算，形成 128 比特位的密码—帐号摘要，该摘要填充到身份验证信息中，为内容 2；

产生一个 128 比特位的随机数字(Rm)，利用该随机数字、上述的密码摘要

以及当前的日期值再合并进行散列函数运算，形成 128 比特位的密码—随机数—日期摘要，该摘要填充到身份验证信息中，为内容 3;

将上述的随机数字(Rm)直接填充到身份验证信息中，为内容 4;

4、如权利要求 3 所述的方法，其特征在于认证服务器利用身份验证信息中的 4 项内容完成对签名人的身份合法性验证，步骤如下:

利用权利要求 3 中提及的内容 1，服务器可以确定需要验证的帐号;

利用权利要求 3 中提及的内容 2，服务器可以从密码数据库中解密出该帐号对应的密码摘要;

利用数据库解密得到的密码摘要、利用权利要求 3 中提及的内容 4、利用当前的日期，服务器可以重新计算密码—随机数—日期摘要;

如果计算得到的密码—随机数—日期摘要和网络传送过来的密码—随机数—日期摘要（权利要求 3 中提及的内容 3）完全相等，则表明签名人的身份是合法的，可以看出，这里的合法性定义就是签名人知道该帐号对应的密码原文或密码摘要。

5、如权利要求 3 所述的身份验证信息的形成步骤，其特征在于通过密码原文计算密码摘要使用了加强模式，步骤如下:

签名人从键盘输入密码原文的一部分，该部分比较短，比较容易记忆;

签名人从特定存储介质中输入密码原文的另一部分，该部分是一个位数很长的随机数，很难记忆;

上述的特定存储介质可以是硬盘文件、软盘文件或 IC 卡等等;

将上述的两部分密码原文合并进行散列函数运算，形成 128 比特位的密码摘要;

密码摘要用于形成权利要求 3 中所述内容 2 和内容 3。

6、一种称为动态签名的单向身份识别技术，其特征如下:

签名人按权力要求 1 所述的方法完成一次签名，该签名人成为身份识别过程中的甲方，也就是被识别的一方;

上述签名的特殊性在于：认证服务器存储到签名数据库的签名记录并不是某个特定源信息形成的数字摘要，而是利用权利要求 3 中提及的内容 4 以及权力要求 4 中提及的数据库解密得到的密码摘要合并进行散列函数运算后得出的秘密摘要。这种由秘密摘要形成的签名记录称为动态签名记录;

上述秘密摘要的特点在于：该摘要没有通过网络层传输，该摘要只有签名人自己和认证服务器能通过计算方法得出;

签名人用上述的秘密摘要加密某种特定的请求信息，然后将加密后的信息发送到身份识别过程中的乙方，一般是提供某种服务的网络服务器;

上述的乙方按权力要求 2 所述的方法进行一次签名验证。该签名验证的特殊性在于：乙方可获得只有甲方才知道的秘密摘要，从而解密出甲方传送来的服务请求信息，认证服务器对这种动态签名记录只允许验证一次。

7、一种通过认证服务器传递一个加密密钥的方法，利用该方法可以实现电子信息的加密传送。该方法包括密钥发送和密钥收取两个过程:

密钥发送过程如下:

密钥发送方（甲方）按权力要求 6 所述的方法完成一次动态签名，然后随机产生一个加密用的密钥 Kr;

用动态签名得到的秘密摘要加密上述的密钥 Kr，然后将动态签名标识信息以及经过加密的 Kr 发送到接收方（乙方）帐号所在的认证服务器;

认证服务器通过验证甲方的动态签名可实现两个目的: 首先是可以确定甲方的真实身份，其次是可以用验证服务得到的秘密摘要解密出 Kr;

认证服务器将 Kr 存储到乙方的密钥数据库中，并向甲方返回一个用于唯一识别 Kr 的密钥序列号;

甲方用 Kr 加密源信息，并将乙方帐号、密钥序列号等识别信息附加到已经加密的信息中。这就形成了一个完整的加密信息。

密钥收取过程如下:

甲方可以通过任何非秘密通道将加密信息发送到乙方手中;

乙方从加密信息中提取密钥识别信息;

乙方将自己的身份验证信息（如权力要求 4 或权力要求 5 所述）及上述的密钥识别信息通过网络发送到自己帐号所在的认证服务器，该过程称为密钥收取请求;

认证服务器通过上述的身份验证信息判断该密钥收取请求是否合法，如果合法，则通过密钥识别信息从密钥数据库中查询得到 Kr，再将 Kr 返回到乙方;如果身份验证信息非法，则返回一个错误信息;

乙方利用认证服务器返回的 Kr，可以从加密信息中解出源信息。

8、一种电子公章的实现方法，该方法包括以下步骤:

一个单位能实现电子公章的前提是在网络中的一台认证服务器上建立一个属于本单位的公司帐号，公司帐号的特殊性在于: 帐号本身不设立密码，签名权限通过一个称为职员列表的数据库进行控制;

加盖电子公章的信息首先由该公司的合法职员产生，这里所说的合法性是指该职员的个人签名帐号已经被加入到公司帐号的职员列表中;

该合法职员首先用一个散列函数计算某特定信息的数字摘要，该特定信息称为源信息;

该合法职员按权力要求 6 所述的方法完成一次个人动态签名;

该合法职员将动态签名标识信息以及经过源信息的数字摘要发送到本公司帐号所在的认证服务器，该过程称为加盖电子公章请求;

认证服务器首先通过验证该职员的动态签名确定其真实身份;

认证服务器然后在公司帐号对应的职员列表数据库中查询该职员的个人帐号信息，如果查到，表明该加盖电子公章请求是合法的，再将传送来的数字摘

要输入到签名数据库中，同时还要记录该签名的形成日期和时间，然后向客户端返回一个可唯一识别该签名记录的签名序列号；如果在公司的职员列表中查不到发送请求的个人帐号，则返回一个错误信息；

盖章职员把上述的签名序列号、公司的帐号名称以及认证服务器的域名（或者是该服务器的 IP 地址）等标识信息附加到源信息中，合并后的信息就成为加盖电子公章的信息；

电子公章的验证过程和权力要求 2 所述的完全一样，只是认证服务器在返回的信息中附加了公司的相关资料。

9、服务端数据库自我安全防护技术，该技术的必要性及实现方法如下：

权力要求 1 至权力要求 8 所述的各项功能及优点，都是通过装载于认证服务器上的服务软件和装载于客户机的客户软件在网络通讯条件下，利用特定的接口协议实现的。本发明的服务软件使用了工业标准的 SQL 数据库查询语言，可以连接各种外部数据库引擎实现数据的存储；

在上述实现机制下，存在这样一个问题：一旦数据存储到外部数据库中，数据的安全性就不再受服务软件的控制，可能被篡改或者被读取，尤其对于管理人员，这种不安全因素更是无法避免；

为了克服上述的不安全因素，本软件使用了一种称为安全密钥组自加密及自签名技术；

所谓的自签名技术可达到下述目的：存入外部数据库的安全数据一旦被非法修改，服务软件再次使用这些数据时，肯定会查觉。这里非法修改的定义就是有关人员绕过服务软件的安全规则设置，企图通过直接修改数据库达到某种目的；

所谓自加密技术其目的在于：所有存入数据库的保密信息（例如帐号密码、帐号之间传递的密钥）都不可能被非法读取。这里非法读取的定义就是有关人员绕过服务软件的安全规则设置，企图直接从数据库中的获得他本来没有权力知道的信息；

上述技术的核心方法就是把存储到外部数据库的每一条记录都使用一次一密的方法进行对称加密处理。

通过上述技术，本发明建立的服务体系，其安全性完全自主实现，不再依赖于外部数据库的安全规则。

00·10·24

# 说　明　书

## 通用电子信息网络认证系统及方法

目前互联网(Internet)的应用已经逐步扩展到社会生活的各个领域，虽然在各国的发展程度有巨大差别，但它的发展趋势是不可逆转的。

随着互联网向人们日常工作、生活的逐步渗透，“互联网虚拟社会”这个词在我们生活中出现的频率也越来越高了，甚至它已经成为了我们实际生活中的一部分。那么，什么是真正的互联网虚拟社会呢？简言之，就是人们日常工作和生活的绝大部分信息通过互联网完成传输，可以基本摆脱纸张的使用，因此互联网虚拟社会也可以称为无纸化社会。

分析当前的网络信息技术可以看出，实现无纸化社会的很多关键技术已经解决了：在现代图形化操作系统及多媒体技术的支持下，计算机已经可以表达任何文字信息，可以实现任何静态或动态图象的数字化储存、传输和再现，可以实现任何声音的数字化储存、传输和再现。这说明，在现有技术支持下，人类社会中的绝大部分信息都可以完成数字化表达和传送。

互联网虚拟社会的完全实现可能还需要较长的时间（至少 10 年以上），但它的初级阶段已经开始进入我们的日常生活，这表现在很多方面：电子邮件开始取代传统的纸笺信；人们日常所需的很多信息（新闻、影视、招聘、商品、技术……）开始通过浏览网页来获取，传统印刷的报刊、杂志、书籍已经让出一席之地了；电子商务的发展虽然曾经遭遇到很多困难，尤其在国内的发展还很不成熟，但它已经为我们提供了 “足不出户尽购天下物”的消费概念。种种迹象表明我们已经处于互联网虚拟社会的雏形阶段。

是不是人类完全进入互联网虚拟社会仅仅剩下一个时间早晚的问题，所有与之相关的技术问题都已经解决了呢？本发明人认为并非如此，目前的当务之急是解决互联网信息安全问题！应该说，信息安全已经成为人类实现互联网虚拟社会的最大障碍。这个问题在政府、国防领域更是突出，这些领域日常处理的绝大多数内容是文字信息，而这些信息以目前的技术水平完全可以通过计算机可视信息表达，也完全可以通过互联网传送。实际情况却是令人悲观的，凡是这类重要领域都因为信息安全问题而严格限制互联网技术的应用。

在网络安全领域，基本可划分为两大领域：(1)攻击防护技术；(2)信息的确定性及信息保密技术；这两个领域在很多方面是相辅相承的，但二者的意义是截然不同的。本文主要针对第二方面，即信息的确定性及保密技术。

目前，国际通用的信息认证技术是利用公开密钥算法实现的，实用的服务机构一般称为认证中心（Certification Authority），简称 CA。本发明提出的

方案是基于客户/服务器协议实现信息签名及保密，是一种现实可行、成本更低、功能更强大、可靠性较高的新的信息认证技术。

目前实用的电子信息签名称为数字签名，它的实现过程是这样的：签名人首先要通过认证中心（CA）获得一个基于公开密钥算法的密钥对，该密钥对包括两个密钥，其一称为公钥，另一个称为私钥。所述的公钥是完全公开的，并包含在认证中心颁发的个人数字证书中，所述的私钥是完全秘密的，需要保存在某种安全的存储介质中，并由签名人自己保管。签名人形成数字签名时，首先计算源信息(M)的数字摘要(Dm)，然后用自己的私钥将该数字摘要加密形成摘要密文($D_r$)，$D_r$+M 就形成一个签名信息。而验证方用签名人的公钥解密 $D_r$ 形成 Dm′，重新计算源信息（M）的数字摘要（Dm），如果 Dm′=Dm，则验证通过。根据公开密钥算法的基本原理，上述数字签名的形成和验证过程是完全可信赖的。

在本发明中，首先对电子信息的安全性进行了严格的定义：电子信息安全性包括信息确定性和信息保密性两个方面，信息确定性由 4 项要素构成（信息发布者身份的确定性、信息发布时间的确定性、信息发布后的不可否认性、信息发布后的不可修改性），信息保密性则是为了实现信息内容不被他人获知的要求。应该说，信息确定性是信息安全的基本要求，而信息保密性则是信息安全的特殊要求。如果某种技术方案能全部实现信息安全性的所有要素，那么就称为信息安全性的完备实现。本发明人提出的通用电子信息网络认证系统（UNSS）就是一种完备实现方案，现在任何一项技术都不具有这种完备特征。

本发明实现信息确定性是通过网络签名技术实现的。在本发明中，电子信息的签名称为网络签名，是为了在概念上与公开密钥算法的数字签名进行区分。

实现上述的网络签名包括以下步骤：一个人首先要在网络中的一台专用服务器（也称为认证服务器或 UNSS 服务器）上建立一个属于自己的帐号；生成签名时，签名人通过网络通讯将本人的身份验证信息及数字摘要（Dm）传送到自己帐号所在的认证服务器，这个过程称为签名请求（对认证服务器而言，称为签名服务）。认证服务器通过身份验证信息判断该签名请求是否合法，如果合法，则将传送来的数字摘要输入到签名数据库中，同时还要记录该签名的形成日期和时间，然后向客户端返回一个可唯一识别该签名记录的签名序列号。签名人把服务器返回的签名序列号、自己的帐号名称以及认证服务器的域名（或者是该服务器的 IP 地址）等标识信息附加到源信息中，合并后的信息就成为签名信息。

对上述网络签名的验证步骤如下：验证方可以通过任何手段接收到签名人提供的签名信息；验证方提取签名信息中的标识信息，然后通过网络通讯将标识信息传送到签名人帐号所在的认证服务器，这个过程称为签名验证请求（对认证服务器而言，称为签名验证服务）。认证服务器通过标识信息可以从数据库中查询到相应的数字摘要（Dm）、以及相关的签名日期、签名时间等公证信

息，然后将这些信息返回给验证方。验证方用形成签名时使用的相同散列函数重新计算源信息的数字摘要(Dm')，如果计算得到的数字摘要(Dm')和认证服务器返回的数字摘要（Dm）完全相等，则表明该签名信息是完全合法的，而且在传送过程中没有被篡改过，与此同时，验证者还获得了该签名形成的公证日期和时间。

综上所述，通过本发明建立的签名服务和签名验证服务，完全实现了前文所述的信息确定性的 4 个要素。

本发明对信息保密性的实现是通过密钥传递协议实现的，步骤如下：甲方使用密钥发送服务将一个随机密钥 Kr 发送到乙方的帐号中，乙方则使用密钥收取服务从自己的帐号数据库中取得密钥 Kr。Kr 就成为甲、乙双方共同拥有的秘密，而任何第三方都不会获知 Kr。在上述机制的作用下，甲、乙双方就可以利用 Kr 来交换任何需要保密的电子信息。在本发明建立的认证体系中，甲、乙双方实现密钥传递的前提就是二者都已经在认证服务器上建立了自己的相应帐号。

综上所述，通过本发明建立的密钥传递协议（含密钥发送和密钥收取两部分），完全可以实现信息保密性的要求。

通过阅读以下结合附图所作的描述可以清楚地了解本发明的各种特征和优点，在所说附图中：

图 1 表示通用电子信息网络认证系统对信息安全性的定义；
图 2 表示通用网络认证系统（UNSS）的结构体系；
图 3 表示通用网络认证系统（UNSS）中网络签名的产生过程；
图 4(a)表示 UNSS 系统中身份验证信息在客户端的生成方法；
图 4(b)表示 UNSS 系统中身份验证信息在服务端的检验方法；
图 5 表示通用网络认证系统（UNSS）中网络签名的验证过程；
图 6 表示 UNSS 系统中动态签名的产生及使用方法；
图 7 表示密钥传递协议中的密钥发送过程；
图 8 表示密钥传递协议中的密钥收取过程；
图 9 表示 UNSS 系统的总体安全框架；
图 10 表示 UNSS 认证体系中的通讯层加/解密技术；
图 11 表示 UNSS 服务软件中安全密钥组的产生来恢复原理；
图 12 表示使用安全密钥组进行自加密和自签名的原理；
图 13 示例了 UNSS 认证系统的一个实际应用方案。

本发明可以利用市售计算机系统和技术建立一个电子信息网络认证的综合体系来实施。

参照图 2，该图表示本申请人所述的通用电子信息网络认证系统的结构体系。该体系必须建立在网络（可以是局域网或 Internet）基础上，至少应建立一台认证服务器（也可以是多台），所述的认证服务器就是在已经装载了操作系统（如 Windows NT、Windows98 等）和数据库引擎（如 Access、SQL server 等）的计算机上再安装由本发明所述技术形成的服务软件（UNSS Server）。图 2 所示的客户端，是指任何一台接入网络的计算机，并可以运行由本发明所述技术形成的客户软件（UNSS Client）。

应该指出的是，UNSS 认证体系是通过认证服务器来确认某一特定的电子信息（主要指电子文件）是属于某个现实中可追溯的实体（可称为担保体或 UNSS 客户），担保体可以是一个人，也可以是一个单位。这种担保的可信赖性，在技术上是由本发明所述的各种技术实现的，而在实施过程中，首先需要担保体和认证服务器的服务商建立一种法律上的信任关系，这一过程称为帐号注册。一旦完成帐号注册，担保体就在认证服务器上有了一个和自己唯一对应的 UNSS 帐号，就可以享受 UNSS 认证体系提供的所有服务。

此 UNSS 认证体系中，对 UNSS 客户的要求只有两条，①他必须向为他担保的 UNSS 服务商提供完全真实的身份证明资料，②他必须严格保密自己的帐号密码，该密码是他在 UNSS 体系中保护个人权益的唯一保障，该密码也可由他随时更换。

在 UNSS 认证体系中，帐号有两种类型：个人类型和公司类型。

在 UNSS 服务软件和客户软件的共同作用下，个人类型的帐号可享受以下服务：①个人签名；②修改个人帐号密码；③向其它个人帐号发送密钥；④收取其它个人帐号发送过来的密钥；⑤向其它个人帐号透露自己的私人信息；⑥获得其它个人帐号透露给自己的私人信息；⑦查询自己帐号的必要信息（如当前的签名序列号、最后的签名时间等等）。本申请不排除将来可能增加的其它服务项目。

在 UNSS 服务软件和客户软件的共同作用下，公司类型的帐号可享受以下服务：①电子公章；②建立和修改本公司的经理帐号；③添加和删除本公司的职员列表。本申请不排除将来可能增加的其它服务项目。

在 UNSS 服务软件和客户软件的共同作用下，任何上网的计算机用户可享受以下服务：①验证某个电子信息的网络签名；②验证某个 UNSS 帐号的类型。本申请不排除将来可能增加的其它服务项目。

以上所有服务都是通过这种机制完成的：客户端软件在用户的操纵下创建服务请求数据，然后将请求数据通过网络发送到相应的认证服务器，服务软件对收到的请求数据作出相应的处理，然后将结果数据再通过网络返回到客户端软件，客户端软件最后向用户显示出相应的结果。上述的过程可以用“请求发送→服务处理→结果返回（接收）”简单表述之。

下面将对 UNSS 认证体系提供的各种服务分述。

关于“个人签名”，参照图 3，该图已经很清楚的显示出一个 UNSS 个人签名的形成过程。首先在客户端计算源信息的数字摘要（Dm）并形成签名人的身份验证信息，然后通过网络向服务器发送签名服务请求，服务器进行判断和处理，再由网络向客户端返回结果。这里需要强调指出的是：完成个人签名必须向认证服务器提供正确的身份验证信息，如果身份验证信息错误，只能得到一个错误信息；此外，完成一次个人签名，在网络层会出现 2 次数据流的传送，第一次是客户端向服务器发送请求数据，第二次是服务器向客户端回送结果数据。

在上述个人签名过程中，最重要的安全数据就是身份验证信息，图 4(a)表示了身份验证信息在客户端的生成方法，图 4(b)表示了身份验证信息在服务端的检验方法。使用本发明所创立的身份验证机制有如下特征和优点：

①单独使用服务器上的密码数据库，不可能获得帐号密码，因为身份验证信息中的内容 2（密码—帐号摘要）是通过数据库求解密码摘要(Pm)的必要条件。而“密码—帐号摘要”本身不会在服务端或客户端的任何介质中存储，只在发生签名服务请求时即时生成，用后及时消毁。

②身份验证信息不可重复使用，因为在服务端有这样一个判断：如果在相同日期条件下，重复使用身份验证信息中的内容 4（随机数 Rm）将是非法的。如果日期条件改变，Rm 当然可以重复使用，但是如果不知道密码摘要(Pm)，也不可能计算出日期改变后的“密码—日期—随机数摘要”，这样，在服务器上仍然检验出错误结果。这里所说的日期条件是现实世界的实际日期，它的标准值是由服务器确定的，如果客户端使用了不真实的日期，也不可能通过身份验证。

③从技术上而言，对密码摘要(Pm)的最佳攻击手段就是在网络层截获一份身份验证信息，再合并帐号名称用穷举法计算“密码—帐号摘要”。

④从图 4(a)可以看出，密码摘要(Pm)是一个 128 比特位的数字，本身具有足够的抗攻击能力，但是该值却直接依赖于密码原文计算得到，因此，真正的安全性依赖于密码原文的复杂性。

⑤如果用户习惯记忆一个比较复杂的密码，密码摘要(Pm)就具备了较高的抗攻击能力；如果用户不习惯记忆复杂密码，他应该使用本发明提供的密码加强模式，将一部分随机生成的复杂密码保存在某种存储介质中，自己再记忆一部分比较简单的密码。

⑥密码加强模式和普通模式可以根据用户的意愿随时转换。

⑦使用该机制，还可提供另外一个随机密钥，即密码摘要(Pm)合并随机数(Rm)进行散列运算得到数字摘要值，本发明中称该值为协议密钥，用 Pk 表示。协议密钥的特征在于：不在网络层进行传送，客户端和服务器分别通过计算方法得到，而且每次通讯都在随机变化。

⑧上述的协议密钥（Pk）可用来加密身份验证信息以外的其它安全数据。在图 3 所示的签名请求数据中，源信息的数字摘要(Dm)实际上已经用 Pk 进行了加

密，这样可以有效地防止 Dm 在网络传送过程中被非法篡改。

关于“验证某个电子信息的网络签名”，参照图 5，该图很清楚的显示出如何验证一个 UNSS 网络签名。客户端首先从签名信息中提取出签名标识信息，然后将标识信息通过网络发送到相应的认证服务器，这个过程称为签名验证请求。认证服务器根据必要的标识信息，从数据库中查询得到相关的信息（包括数字摘要 Dm、签名日期、签名时间等等），然后再通过网络把这些信息返回到客户端。客户端再从签名信息中提取源信息，并重新计算其数字摘要（Dm'），如果 Dm'和服务器提供的 Dm 完全相等，就可以得出以下结论：该信息的发布者身份是确定的、发布时间是确定的、不会被否认、发布后没有被修改过。这里需要强调指出的是：UNSS 的签名验证服务不需要客户端提供任何身份验证信息，只需提供必要的签名标识信息（签名人帐号和签名序列号）；此外，完成一次签名验证，在网络层会出现 2 次数据流的传送，第一次是客户端向服务器发送请求数据，第二次是服务器向客户端回送结果数据；再此外，网络签名比数字签名的最大优势在于它提供了签名的公证日期和时间，而且网络签名具有永久可验证性。

在理解 UNSS 签名机制及签名验证机制的基础上，下面将阐述 UNSS 体系提供的动态签名功能，该功能主要用于单向身份识别。图 6 是动态签名的实现机制示意，这里需要指出的是：所谓的单向身份识别是指图中乙方可以信赖的识别出甲方的身份，而甲方并不能确定乙方的身份。还需要指出的是：动态签名一般涉及三方关系（甲、乙及认证服务器），一般包括 6 次不同时序的网络层数据流传送。可以看出，动态签名的本质仍然是应用了 UNSS 系统提供的签名及签名验证协议，但与前述的签名及签名验证还是有较大差别，其特征如下所述：

特征之一为：在签名实现过程中，客户端并不向服务器传送某个源信息的数字摘要，实际上只传送了身份验证信息，而服务端记录到数据库中的签名摘要则是通过身份验证信息计算得到的，是一个秘密摘要 Ds，实际上该 Ds 就是前文中已经阐述过的协议密钥 Pk。

特征之二为：秘密摘要 Ds 只允许被验证一次，在实现技术上，Ds 被消毁的可能有三种，一是被验证了一次、二是该帐号又进行了一次新的动态签名、三是一定时间后自然消毁。

特征之三为：一个帐号在同一时刻只能存在一个有效的动态签名，该特征实际上是特征之二的自然导出特征，此外，由于该特征的约束，乙方在验证动态签名（获得 Ds）时不需要提供签名序列号，只提供帐号名称就足够了。

特征之四为：甲、乙双方可以按照约定协议用 Ds 建立单向信任关系。对于图 6 还应该说明的是：图中的乙方可以是任何需要识别甲方身份的网络结点，甚至可以是另一个 UNSS 认证服务器。

关于“向其它个人帐号发送密钥”，该过程通过 UNSS 系统提供的密钥发送协议得以实现，图 7 表示了这一过程。分析该图不难看出，密钥发送实际上就

是所述动态签名的一个具体应用，这里的认证服务器 A 相当于图 6 中的认证服务器，这里的密钥发送者相当于图 6 中的甲方，这里的认证服务器 B 则相当于图 6 中的乙方。因此，理解了动态签名机制也就很容易明白密钥发送的实现机制。此外，需要指出的是：认证服务器 A、B 可以是同一台 UNSS 服务器，在这种情况下，图中所示的④和⑤将在服务器内部实现，不经过网络层传送。再一点还应该指出：认证服务器 B 在存储 Kr 的同时，还会标记下密钥发送者（甲方）的真实身份，根据动态签名协议，这种身份标记是可信赖的。

关于“收取其它个人帐号发送过来的密钥”，这一过程的实现利用 UNSS 系统提供的密钥收取协议完成，参见图 8。从图中可以看出，客户端必须提供正确的身份验证信息才能完成密钥收取，关于身份验证信息的使用方法前文已经做过阐述。此外，传递密钥 Kr 由服务器传送至客户端是经过加密的，加密用的密钥是本次身份验证信息计算出的协议密钥 Pk，关于 Pk 的实现方法请参见前文。还应该指出：客户端在收到 Kr 的同时，还能获得 Kr 发送者的真实身份。

在理解上述密钥发送和密钥收取机制的前提下，还应强调指出：通过这一对协议， UNSS 认证体系可以提供双向的身份识别技术。其理由如下：密钥发送者（甲方）一旦成功发送密钥 Kr，他的真实身份就会与 Kr 同时传递给密钥收取者（乙方），这一点与动态签名有类似之处，但二者最大的区别在于：获取动态签名 Ds 时，不验证乙方的身份；而获取传递密钥 Kr 时，乙方必须提供正确的身份验证信息。这就表明，甲、乙双方可以通过 Kr 建立相互信任的秘密信道，因为二者的身份都是确定无疑的。此外，还需要指出的是：与动态签名提供的单向身份识别相比，实现双向身份识别要增加的代价有两点，①乙方必须建立自己的 UNSS 帐号、②网络层的数据流传送多了 2 次。

关于“修改个人帐号密码”，该过程概念上比较简单，不再用图解说明。其特征在于：更新帐号密码必须提供基于当前密码的身份验证信息，验证信息的生成方法如前所述；新密码摘要在送往服务器之前，使用本次协议密钥 Pk 进行加密；在客户端，用户可以自主选择新密码是否使用如前所述的加强模式。

本文至此，应该特别指出，前述的个人签名（包括动态签名）、密钥收取以及修改密码是 UNSS 认证体系中仅有的 3 项核心级服务，其特征在于：①服务请求数据直接发送到本帐号所在的认证服务器；②服务请求数据必须提供正确的身份验证信息；③服务过程仅涉及 1 个帐号、1 台认证服务器和 2 次网络层数据流传送。而前述的签名验证（包括动态签名验证）是 UNSS 认证体系中最为重要的 1 项公开级服务，其特征在于：①服务请求数据不需要提供身份验证信息，只提供必要的标识信息；②服务过程仅涉及 1 个帐号、1 台认证服务器和 2 次网络层数据流传送。而前述的密钥发送以及后面还要阐述的所有其它服务，都属于 UNSS 认证体系中的衍生级服务，其特征在于：①身份识别使用了前述的动态签名技术；②服务过程可能涉及到 2 个帐号和 2 台认证服务器；③网络层数据流传送可能需要 4 次或 6 次。

关于“向其它个人帐号透露自己的私人信息”，该项功能是通过 UNSS 提供

的私人信息透露协议实现的，其特征如下：甲方用户和认证服务器之间使用动态签名技术完成身份识别，具体过程可参考图 7，需要强调的是认证服务器 A 和 B 肯定是同一台服务器；在服务请求数据中，甲方需要指明乙方的 UNSS 帐号和透露内容的信息屏蔽字，所述的信息屏蔽字是一个 32 或 64 比特位的整数，通过比特位上的 0 或 1 确定对应的某项私人信息是否透露；服务器在身份识别正确的前提下，将乙方帐号和信息屏蔽字存入相应数据库。此外，也应该指出：私人信息透露协议可以通过直接提供身份验证信息实现，因为该过程实质上只涉及甲方帐号所在的服务器，不会涉及到 2 台以上的认证服务器，如果直接提供身份验证信息，该协议就转化为核心级服务了，这无疑会加大安全代码的用量，出现安全漏洞的可能性也会增加，基于这种考虑，私人信息透露协议仍然以衍生级服务的方式实现。

关于“获得其它个人帐号透露给自己的私人信息”，该过程是通过 UNSS 提供的私人信息获取协议实现的，其特征如下：甲方已经利用前述的透露协议向乙方透露了某些私人信息；乙方用户使用动态签名请求甲方帐号的认证服务器进行身份识别；服务器在身份识别正确的前提下，根据乙方帐号以及相应的信息屏蔽字将甲方的私人信息通过网络返回给乙方。需要指出的是：私人信息获取协议必须实现为衍生级服务，因为该过程可能涉及两台认证服务器。

如前所述，UNSS 认证体系提供的私人信息透露及获取协议提供了一种比较简单、容易理解、有一定安全性的私人信息共享技术，其理由如下：甲、乙双方以认证服务器为中介就可以交换完全可靠的私人资料（真实姓名、私人电话、性别、年龄等等），这些资料是用户在帐号注册时提供给 UNSS 认证服务器的，而且不可能私自修改，只能通过认证服务器的管理人员进行有条件的修改，因此通过这种方式获得的私人资料决不可能有虚假信息；甲方可以完全自主决定向乙方透露哪些内容，这无疑对保护个人隐私权是相当有利的。

关于“查询自己帐号的必要信息”，这是一个很简单的服务，用户通过动态签名向服务器证明身份，服务器返回帐号的相关状态，如当前的签名序列号、最后的签名时间等等。需要强调的是：该项功能可以实现为核心级服务，但基于安全代码用量角度考虑，仍然以衍生级服务方式实现。

关于“电子公章”。 公章的实质是一种团体签名行为。我们看一下在现实社会中如何盖一个有效的公章：张三是 A 公司的职员，他要发送一封订货公函，他首先要添写一个公函存根（可能还需领导签字），公函存根由公章室存档后，再为订货函加盖单位公章。在这个过程中，公章室的真正职能是存档，而实际的盖章人是张三，如果该公函日后发生纠纷，可追溯的第一负责人是张三，并不是公章室。当然 A 公司的其它职员也有类似的工作权力。

应该说，公章的社会信誉是以该单位的团体信誉保证的。但是，在团体内部，每个公章文件又必须有可追溯的个体负责人（上例中就是张三先生）。也就是说，社会的根本信誉仍然是个体信誉，这一社会属性，我们可称为个体信誉支配原则。

根据上述分析，实现电子公章，必须做到以下两点：

① 对外应该体现出团体特性

② 在追溯需求下，应该能查到个体负责人。

电子公章在公钥方式的数字签名体系中，实现起来十分困难；而在 UNSS 认证体系中，却可以很容易的实现，步骤如下：首先将某个 UNSS 个人帐号添加到某个 UNSS 公司帐号的职员列表中；该职员使用动态签名技术向公司帐号所在的认证服务器证明自己的身份；服务器判断该职员帐号是否存在于相关公司的职员列表中；在身份真实及属于合法职员的双前提下，认证服务器就会把服务请求数据中的信息摘要记录到相关公司帐号的数据库中，并返回一个签名序列号。电子公章的验证过程和私人签名的验证过程完全一样，只是服务器返回的信息有所不同，电子公章验证后，服务器除了返回信息摘要、签名时间等基本资料外，还会提供公司的有关资料（如公司名称、公司电话、公司地址等等）。

关于“建立和修改本公司的经理帐号”，所谓经理帐号是一个已经存在的 UNSS 个人帐号，在注册公司帐号时必须同时提供一个经理帐号，而一个公司帐号在同一时间只允许有一个经理帐号。经理帐号有如下权力：首先是添加或删除本公司的职员列表；其次是经理帐号直接具备加盖电子公章的权力，无需添加到职员列表中；再其次是更换经理帐号本身，一旦完成更换，这两项权力就转交给新经理了，旧的经理帐号就失去所有权力了。上述的 3 项权力都是通过动态签名技术实现的，都属于 UNSS 认证体系中的衍生级服务。

关于“添加和删除本公司的职员列表”，如前所述，这项功能是公司之经理帐号所具备的一项权力，通过动态签名技术实现。

关于“验证某个 UNSS 帐号的类型”，这是 UNSS 认证体系提供的一项辅助性的公开级服务，其特征在于：客户端无需提供任何身份识别信息，可以自由查询某个 UNSS 帐号的类型（如个人类型、公司类型等）。

至此，UNSS 面向用户的所有服务功能都已经阐述完毕。

下面将阐述 UNSS 认证体系的安全技术，图 9 表示 UNSS 系统的总体安全框架，该图以概要方式显示了 UNSS 认证体系中的所有安全层次。可以看出，UNSS 体系的安全层次有 3 个：(1)通讯层加/解密；(2)协议层加/解密；(3)存储层自我防护（仅对服务端而言）。

这 3 个加密层的特征首先在于：（1）和（2）是确保客户端和服务端连接的安全性，双方必须遵守完全相同的加/解密规则，因此（1）和（2）也可以合称为连接层加/解密，也就是通常所说的端到端的加密方法。存储层自我防护（3）是服务端为了确保外部数据库中存储数据的安全性，所用规则是单方面的，与客户端没有任何关系。

所述特征其次在于：（1）和（2）表示的每一个完整过程都是以两次网络层数据流为单位的，即，客户端在用户的支配下填充请求数据块，该请求数据首先经过协议层（2）的加密，然后经过通讯层（1）的加密，最后形成 1024 字节的密文数据，该密文数据经由网络传送到服务端，服务端按相同顺序、相同

规则再进行解密，服务端返回结果时，也首先进行协议层加密，再进行通讯层加密，最后发送，客户在接收到结果后，再按相同顺序及规则进行解密，至此，一个完整的连接层加解密过程就完成了。可以看出，所述的两次网络层数据流就是指客户端产生的请求数据流和服务端产生的服务结果数据流。

下文对图 9 表示的 3 个安全层分述之。

首先阐述通讯层加/解密技术，该层的实现采用了类似安全套接字（SSL 协议）的技术，图 10 表示了该层安全技术的数据结构，其特征在于：每个 UNSS 认证服务器应该有至少一对 RSA 公/私密钥对，其中的公钥由 UNSS 信任中心统一保管，而对应的私钥只存在于认证服务器内部。客户端在与某个 UNSS 认证服务器通讯之前，首先应该通过 UNSS 信任中心获得该服务器的一个公钥 Upk，然后随机产生一个 128 比特位的通讯密钥 Ckr，并用 Ckr 加密本次服务请求数据，再然后用服务器的公钥 Upk 加密 Ckr，形成密文 M(Ckr)。对应认证服务器接收到完整的数据包（1024 字节）后，用 Upk 对应的私钥 Usk 把 M(Ckr)解密为 Ckr，再用 Ckr 解密本次服务请求数据。应该强调的是：通讯层加/解密对任何 UNSS 客户/服务器通讯采用完全相同的算法，区别仅在于 Upk 和 Ckr 不同，Upk 是和特定认证服务器相关联，而 Ckr 则是随机产生的。还应指出：对于服务请求过程，该层的加密发生在客户端（包括一次对称加密算法和一次 RSA 公钥加密），发生在任何服务请求数据填充完成之后，网络传送之前，而对应的的解密发生在服务端（包括一次 RSA 私钥解密和一次对称算法解密），发生在网络数据接收之后，任何服务开始之前；对于服务结果回送，该层的加密发生在服务端（仅包括一次对称算法加密），而对应的解密发生在客户端（仅包括一次对称算法解密）。

通讯层加/解密可确保 2 个安全性：①防止发生服务器欺骗；②防止任何利用网络数据流进行服务类型分析。根据 RSA 公开密钥算法的安全性，以上 2 点在私钥未被攻破的前提下是完全可以信赖的。

下面阐述协议层加/解密，该层加密根据服务类型的不同有着不同的特点，在前述的各项服务中已经作过一些说明，其特征在于：对于 UNSS 认证体系的 3 项核心级服务（个人签名、密钥收取、修改密码等），每次服务的加/解密密钥使用对应的协议密钥 Pk。个人签名服务 Pk 用来加/解密信息摘要 Dm（客户端加密，服务端解密）；密钥收取服务 Pk 用来加/解密接收密钥 Kr（服务端加密，客户端解密）；修改密码服务 Pk 则用来加/解密新密码（客户端加密，服务端解密）；动态签名略有些特殊，Pk 直接作为秘密摘要 Ds 在服务端记录到数据库中。对于 UNSS 认证体系的所有衍生级服务，统一用动态签名摘要 Ds 加/解密，其安全性前面已经作过阐述。对于 UNSS 认证体系的所有公开级服务（签名验证、查询帐号类型等），不存在协议层加密，其安全性将完全依赖前述的通讯层加/解密。还应指明的是：协议层加/解密只使用对称密钥算法。

协议层加/解密的作用在于：①在发生服务器私钥被攻破的情况下，确保 UNSS 服务仍然具备一定的安全性，特别是确保最敏感信息（如帐号密码）仍然相对

安全。②在网络环境比较安全的前提下（比如小型的局域网中），可以适当的取消通讯层加/解密保护，以提高服务响应速度，因为 RSA 非对称加/解密算法比较耗费时间。

对上述的连接层加/解密技术还应强调指出：对于 UNSS 认证系统的 3 项核心级服务，协议层加/解密可以提供相对安全的数据保护，因此对通讯层加/解密的依赖性较弱；对于 UNSS 认证系统的公开级服务，由于不能提供协议层加/解密保护，其安全性只能依赖于通讯层加/解密；而对于 UNSS 认证系统的衍生级服务，如果服务过程只涉及一台认证服务器，协议层加/解密也可以提供相对安全的数据保护，如果服务过程涉及了两台认证服务器，那么在两台服务器之间必然会发生一次签名验证过程，其安全性只能依赖通讯层加/解密了。基于上述原因，UNSS 认证服务器之间的服务连接（必然是一次动态签名验证的服务连接）必须强制使用通讯层加/解密，以确保整个认证体系的安全性。

下面阐述存储层自我防护技术，该项技术与权力要求 9 是相对应的。使用存储层自我防护技术的原因在于：UNSS 认证服务器的所有数据都是存储于外部数据库，服务软件通过工业标准的 SQL 语句访问数据库，因此 UNSS 认证服务器可以根据实际条件使用任何软件商的数据库引擎（如 Access、SQL server 等等）。基于上述机制，UNSS 认证服务器必须考虑解决外部数据库的数据安全问题，比如直接关系用户利益的安全问题有：帐号密码是否能被他人获取？本人的信息签名是否会被他人篡改？保存于服务器的传递密钥是否可被他人读取？应该指出的是：任何安全问题都不可能是绝对的，都是相对概念上的安全；就上述举出的 3 个问题而言，实际上已经具备基本的安全性了，首先是服务器的数据库不太可能被攻击者轻易接触，其次可以使用外部数据库自身的安全规则，比如规定用户及密码、限制访问权限等等。但是，应该看到，这种外置的安全性是极其脆弱的：首先就是各种数据库引擎的安全级别和安全规则可能差别很大，其次对于服务器的维护人员，由于工作的需要，不可能作到粒度很细的安全约束，即使做出规定，恐怕也不容易实施。

鉴于上述情况，UNSS 服务体系对数据库信息必须实现内置安全性，也就是说，系统的存储安全性要达到这样的目的：①不依赖于外部数据库的安全特性，外部数据库的信息甚至可以随意公开；②由管理人员导致的安全风险应该降到最低限度。

为实现上述目的，UNSS 服务软件使用了本发明提出的数据库自我防护技术，所述的自我防护技术包括自加密和自签名两项内容。

实现自我防护的第一步是生成一个规定大小的安全密钥组，可以从 512 字节～几 K 字节，UNSS 服务软件的密钥组大小为 1024 字节。所述密钥组按图 11 所示步骤产生和保存：首先设定系统的安全员数量（至少 2 人，至多为 10 人），比如 3 名，各安全员按次序输入本人的标识符和密码（比如，张三：P985KQ，李四：357^PL，王五：MK99=321），在输入过程中，系统将建立 2 个初始字符串，一个是真随机串，其特征在于：字符串由人为不可控数值构成，比如键盘

敲击时间、系统时钟、线程句柄等等，而且几乎不可能再现；另一个是伪随机串，其特征在于：字符串由人为可控因素构成（即各安全员标志符和密码的顺序连接），而且可以准确再现。使用上述的真随机串，可以通过散列函数构造1024 字节的真随机密钥组，称 KPA，使用上述的伪随机串，可以通过散列函数构造 1024 字节的伪随机密钥组，称 KPB，然后 KPA 和 KPB 按字节进行异或运算，可以产生第三个 1024 字节的密文密钥组 KPC。用公式表示为：

KPA(真随机密钥组)^KPB(伪随机密钥组)=>KPC（密文密钥组）

KPC 将被导出到存储介质上，如一张软盘（该介质一定要由专人妥善保管）。KPB 当即消毁，KPA 则被锁定在内存中使用。

上述第一步完成后，KPA 就成为访问数据库的安全密钥组，其特征还在于：KPA 只能在内存中生成、恢复和使用，不能导出；正确恢复 KPA 的必要条件有两个：①提供正确的密文密钥组 KPC；②提供正确的伪随机密钥组 KPB。需要指出的是：提供 KPC 必须能获得它的物理存储介质，而获得 KPB 则必须再现前述的伪随机串，也就是说必须由各安全员按原来顺序提供完全一样的标志符和密码，其中任何一个比特位有错误，都不可能正确重现 KPB，也就不可能恢复正确的 KPA。这里的实质在于使用了一种相当安全的密钥多人重构协议。

一般来说，一个数据库只应该和一个安全密钥组相关联，如果一个数据库使用了一个以上的安全密钥组，其结果必然会导致数据库不能被完整地访问。

上述安全密钥组 KPA 将被用自加密和自签名，见图 12 示意。至于是选择自加密还是选择自签名或者同时选择，这要根据数据项的特征作出决定，下面介绍几个最敏感的数据项。

第一是帐号密码，这个数据项必然要选择自加密。由图 12 可知，自加密的实现特征在于：抽取与某关键数据项有固定关系的非关键数据项，比如与帐号密码有固定关系的数据包括帐号名称、帐号 ID、密码的更新时间等等。此外，在 UNSS 服务软件中，帐号密码的加/解密更有其特殊之处，还需要客户端在身份验证信息中提供的“帐号—密码摘要”（见图 4）。这些关联数据项合并为一个密钥种子 Sk，使用 Sk 并通过一定算法就可以从安全密钥组 KPA 中得出一个伪随机的最终密钥 Kf（见图 12），然后用 Kf 加/解密帐号密码，加密发生在写入数据库之前，而解密发生在读取数据库之后。这里需要指出的是：在 KPA 相同的前提下，用不同密钥种子得到相同密钥的概率几乎为 0；在 Sk 相同的前提下，用不同安全密钥组得到相同密钥的概率也几乎为 0；通过密钥种子从安全密钥组中计算最终密钥使用了伪随机置换和散列函数，也就是说，即使知道大量的密钥种子和对应的最终密钥也不可能计算出安全密钥组的构造情况。

第二是帐号之间的传递密钥，该数据项也必须使用自加密，因为该数据项必须严格保证只有接收帐号提供正确身份验证信息后才能获得。为实现自加密，首先也要产生一个密钥种子 Sk，也是通过合并一些有固定关系的非关键数据项产生，如接收帐号 ID、密钥本身的序列号、密钥发送者的帐号名称等等，利用 Sk 及安全密钥组就可得出最终密钥 Kf，传递密钥再用 Kf 加/解密，同样是写入

前加密，读取后解密。

第三是签名形成的信息摘要，该数据项可使用自加密，也可使用自签名，安全性是一样的，但是考虑到动态签名摘要的保密性以及代码实现上的简洁性，本系统还是选择了自加密。自加密用的 Sk 主要由帐号 ID、签名序列号等合并形成。

第四是公司职员列表，该数据项是典型的自签名数据项，因为公司职员不允许直接操作数据库进行添加或删除，只能由公司帐号对应的经理帐号完成这种操作，通过自签名技术，UNSS 服务软件可以轻易判断出是否为非法添加的帐号。从图 12 可以看出，自签名和自加密是非常相似的，最大差别有 3 点：①自签名在产生密钥种子 Sk 时，关键数据项本身必须被合并使用，在本例中，Sk 包括公司帐号 ID、职员帐号名称以及其它标志；②自签名必须增加额外的存储数据项，用来保存由 Sk 产生的最终密钥 Kf；③自签名产生的 Kf 并不用来加/解密，而是直接存入数据库，再次读取和使用相关记录时，可以用来校验数据的合法性。

就上述的自加密和自签名技术，还应该指出：自加密一定程度上隐含着自签名的功能，比如关于传达密钥的自加密阐述，实际上已经具备自签名特性了，因为密钥发送者的帐号名称也不应该被篡改，如果被篡改，将产生不同的 Sk 和不同的最终密钥 Kf，导致的后果就是恢复出一个无效的传达密钥，这一点在服务端虽然不作任何判断，但在客户端却可以作出相应判断和决策。

上述的数据库自我防护技术具有可信赖的安全性，理由如下：通过安全密钥组技术，基本上实现了现代密码技术的最高安全要求，即一次一密钥，因此，通过密码分析技术攻击外部数据库的将是极为困难的。

上述的数据库自我防护技术会带来如下优点：①完全摆脱了对外部数据库安全规则的依赖；②为管理人员的日常维护提供了极大的方便性，他们可以不受任何限制的备份数据或修改某些与安全无关的数据项；③只要做好日常备份工作，系统不会受到毁消性攻击，对此，我们可以作这样的比较：对于 Unix、Windows NT 或其它常规操作系统，攻击者可以通过获取磁盘上的密码数据库进行高效的密码攻击，如果碰巧分析出超级用户的密码，该系统就会不堪一击，而在 UNSS 认证系统中，任何人拿到数据库也只能是望洋兴叹，基本上得不到任何能危害系统安全的信息。同时也应该指出，UNSS 服务端的安全性直接依赖于系统的安全密钥组，也就是依赖于系统安全员的集合安全性（集合安全性的含义是指所有安全员同时集合才可能对系统的安全性造成危害）。

基于上述安全密钥组技术的原理，本发明也不排除用特殊硬件方式实现 KPA 的存储以及通过 Sk 计算 Kf 的全部算法过程，但该硬件的设计应该考虑在多线程环境下使用，应该同时提供多个计算通道。

至此，UNSS 认证体系在实现原理以及在原理支配下的每一个技术环节都已经做了明确阐述。但是，作为一个可操作的完整系统，有必要对 UNSS 的管理模式作进一步的介绍。

作为介绍管理模式的前提，首先应该再明确一下前述的 UNSS 帐号类型。按照 UNSS 的实现原理，帐号类型只有 2 种，其一是个人型帐号，其二是公司型帐号，二者的本质区别在于：个人型帐号通过直接提供身份验证信息完成个人签名服务，而公司型帐号则是通过动态签名技术完成电子公章服务，该类帐号本身没有帐号密码，是通过职员列表实现权限控制的。

在 UNSS 实际系统中，为了应用上的灵活，个人型帐号又被分为 3 种：①自由帐号；①注册帐号；③虚拟帐号。所述 3 种帐号有如下特征：任何用户都可以利用 UNSS 客户端软件在线注册自由帐号，由于自由帐号没经过服务方的真实身份确认，因此只能进行试用，服务方不为该类帐号提供任何信誉担保；如果用户的真实身份经过服务方确认就可以获得一个注册帐号；所谓的自由帐号和注册帐号在使用方法上没有任何区别，在安全方面也没有任何区别，最大的区别就是服务器是否提供信誉担保，此外，在系统规则中，自由帐号允许删除，注册帐号不允许删除。所述的虚拟帐号，属于一种特殊的注册帐号，该类帐号主要提供给某些网络服务器，因为有了 UNSS 帐号的服务器就可以实现前述的双向身份识别，这对某些有特殊安全要求的服务器是相当重要的。还应指出：虚拟帐号不提供“个人签名”服务（允许动态签名），因为虚拟帐号本身是与一台网络服务器对应，并不与现实中的人对应，因此不需要永久签名；根据前文阐述，实现双向身份识别利用了 UNSS 认证体系提供的密钥发送和密钥收取服务，因此禁止永久签名并不影响设立虚拟帐号的目的，也不会影响虚拟帐号的使用。

在定义 UNSS 帐号类型的基础上，UNSS 认证系统对管理功能也进行了十分明确的定义，管理功能总共有 3 项：①为系统添加可担保的帐号（包括个人型注册帐号、虚拟帐号、公司型帐号）；②为个人帐号强制修改帐号密码；③管理人员自身的添加和删除。

基于上述 3 项权力，UNSS 系统又把管理人员划分为 2 种：①管理员；②代理员，可以统称为操作员。其特征在于：管理员允许使用上述 3 种权力的全部；而代理员仅有一种受到限制的权力，其拥有的唯一职能就是向系统添加个人型注册帐号。还需强调指出的是：UNSS 系统上述的管理功能完全利用了系统自身的安全特性，全部使用动态签名技术进行操作员的身份识别，管理权限则通过数据库中的操作员列表进行控制。

上述管理功能用动态签名实现的步骤为：操作员首先向本人帐号所在的认证服务器 A 请求并完成一次动态签名服务；然后向他要操作的认证服务器 B 发送某一具体的操作请求数据，该请求数据使用动态签名之秘密摘要 Ds 加密；认证服务器 B 收到请求后，首先通过签名验证服务从服务器 A 获得秘密摘要 Ds，然后用 Ds 解密请求数据，再然后利用操作员列表判断请求是否符合安全规则，如果请求合法，则完成相应操作。所述的服务器 A 和服务器 B 可以是 1 台服务器也可以是 2 台服务器，无论如何对操作员实现其权力并无影响。可以看出，UNSS 的管理功能，实际上可以归类为 UNSS 的衍生级服务，因为它们具有完全相同的

实现机制，差别主要在于：管理功能要涉及到系统操作员列表、系统事件审核记录等安全控制项目，而普通的衍生级服务不涉及这些项目。

UNSS 系统的上述管理方式有如下特征：①UNSS 操作员必然是 UNSS 认证体系中的一个合法个人用户；②不要求操作员的帐号必须在他所管理的 UNSS 认证服务器上，该特性是目前所有其它操作系统不具备的，其它任何系统的管理帐号必然是本地系统的一个帐号；③上述的 3 种管理功能都是以动态签名技术为基础、以客户端远程方式实现的。④基于前述的 3 个特征，还应指出，UNSS 认证体系实现了真正意义上的分布式管理，系统操作员和服务器并没有本质上的关联性，一个有 UNSS 帐号的个人，可以同时用相同的个人身份成为几台认证服务器的管理员或代理员。⑤UNSS 认证系统对操作员的权限进行了最小范围的明确定义，事实上已经取消了“超级用户”的概念。

上述 3 项服务中的第②项（为个人帐号强制修改帐号密码）需要作如下说明：该管理功能是一项具有潜在危害用户权益的超越功能，但该功能又不得不使用，因为用户忘记密码是可能发生的事件。为了防止管理员恶意使用这种超越功能，系统在核心代码上建立了如下安全规则：强制修改密码必须有两个不同的管理员完成，第一个管理员有权力使一个帐号进入强制修改密码的预备状态，第二个管理员可以对已经进入预备状态的帐号实施密码修改，而且这两个管理员必须不同。此外，这种“预备→修改”操作只有顺序上的限制，并没有时间上的限制，因此两个管理员可以在不同时间、不同地点合作完成这一工作。再此外，无论是预备还是强制修改，在系统安全事件数据库中都有永久的审核记录。

综上所述，UNSS 认证系统在服务功能的基础上，合理、安全地扩展出管理功能，而且将管理功能明确定义为一个最小集合，从而最大限度的控制了安全隐患的发生。服务功能和管理功能合并实现后，整个 UNSS 系统就成为一个可实用的、极为方便的、有很高安全性的在线电子信息认证系统。

由于 UNSS 认证体系提供的信息安全服务是完备的，特别是 UNSS 网络签名具有永久可验证性及自动时间戳，使得 UNSS 的二次开发极为简单。图 13 示意了一种以 UNSS 认证体系为基础的在线银行服务原理。

按照图 13 的示意，一次成功的在线银行业务服务，总共需要 3 次网络连接。对 UNSS 服务器而言，远程客户端和在线银行服务器实际上都属于 UNSS 客户端，只不过一个是申请签名服务而另一个是申请签名验证服务。整个过程在正常网络负载情况下，应该在几分钟之内就可完成。

还应指出：图 13 是一个普适性的服务原理，可以推广到任何需要身份识别或信息确定性判断的实用领域（在线股票交易、在线期货交易、在线竞标、在线拍卖……）。在原则上，任何领域都不会脱离签名/签名验证这一对基本信息安全模式，都应该需要 3 个不同时序的网络连接，而且只需要保证“UNSS 服务端<==> UNSS 客户端”的安全连接就可以了（这种安全性是 UNSS 自身做出保证的），其它任何通道都不需要安全要求。

# 说 明 书 附 图

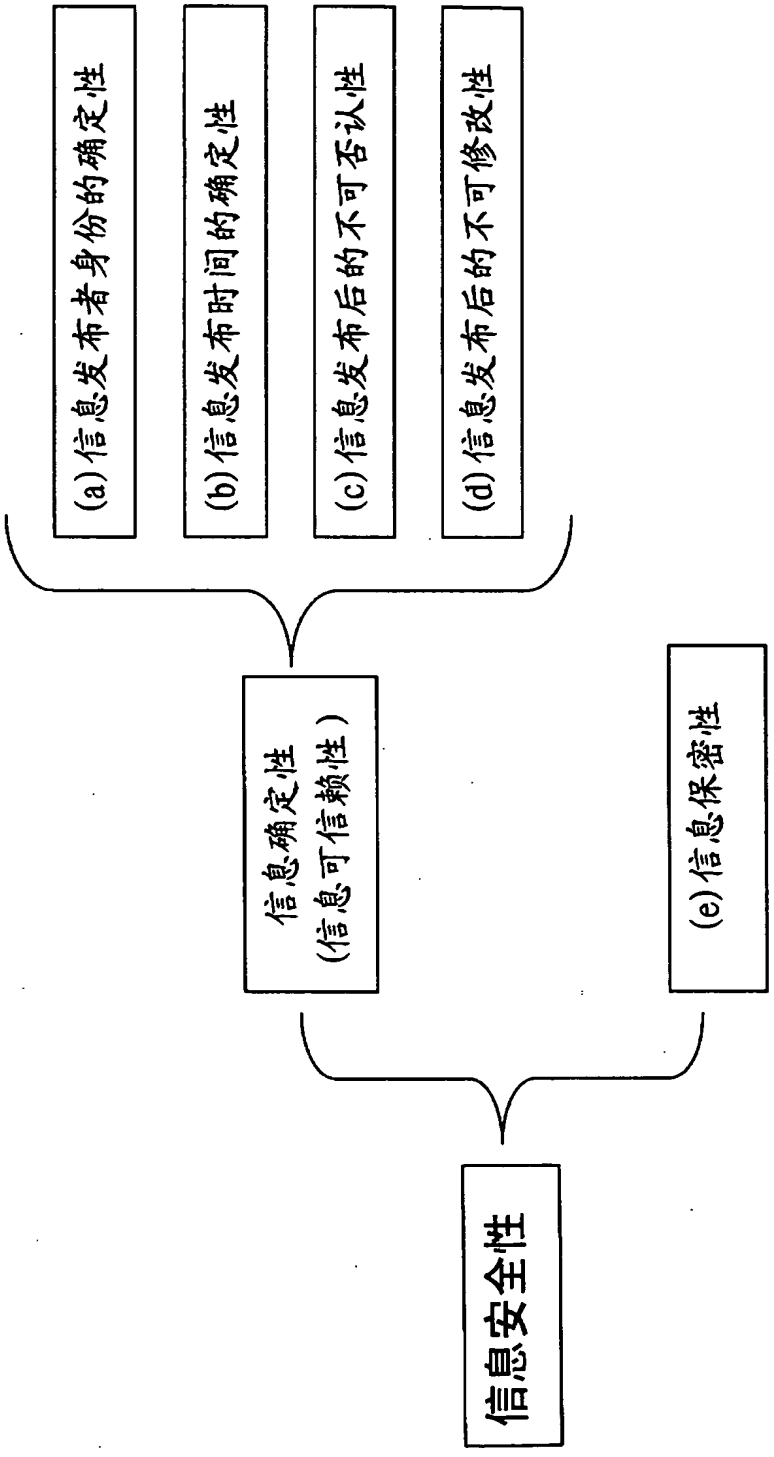

信息安全性
信息确定性
(信息可信赖性)
(a)信息发布者身份的确定性
(b)信息发布时间的确定性
(c)信息发布后的不可否认性
(d)信息发布后的不可修改性
(e)信息保密性

图 1

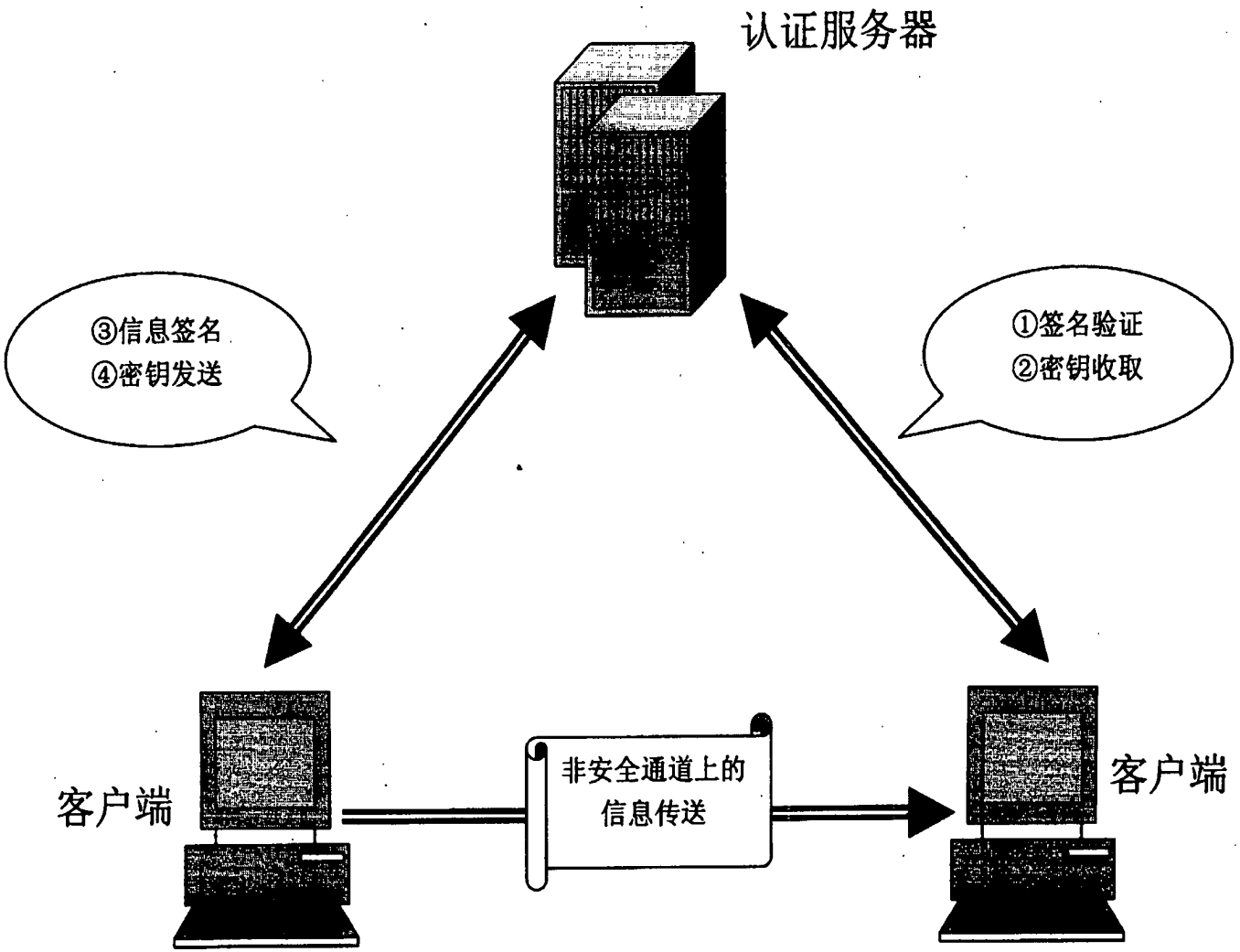



图 2

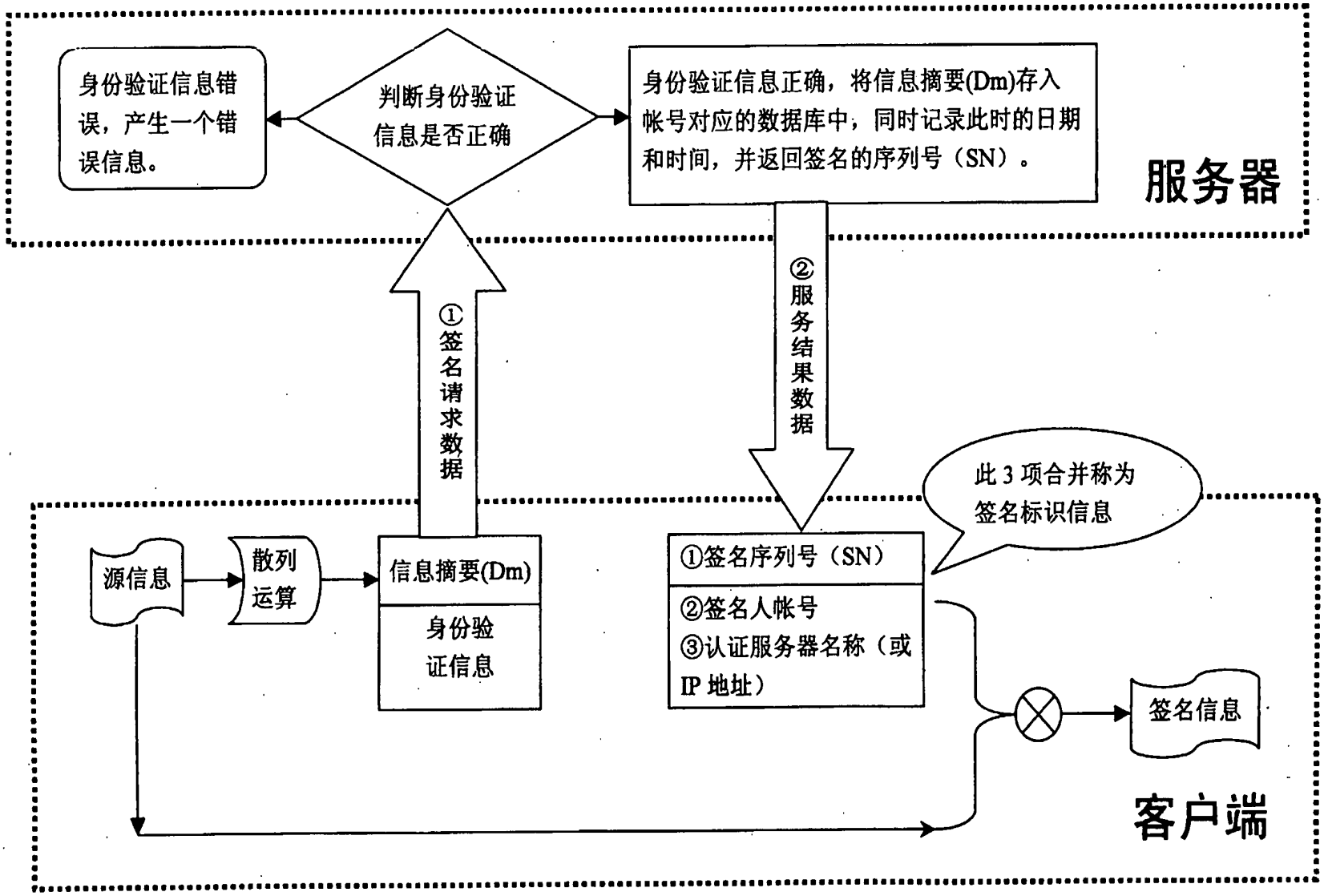

身份验证信息错误，产生一个错误信息。
判断身份验证信息是否正确
身份验证信息正确，将信息摘要(Dm)存入帐号对应的数据库中，同时记录此时的日期和时间，并返回签名的序列号（SN）。
服务器
①签名请求数据
②服务结果数据
此3项合并称为签名标识信息
源信息
散列运算
信息摘要(Dm)
身份验证信息
①签名序列号（SN）
②签名人帐号
③认证服务器名称（或IP地址）
签名信息
客户端

图 3

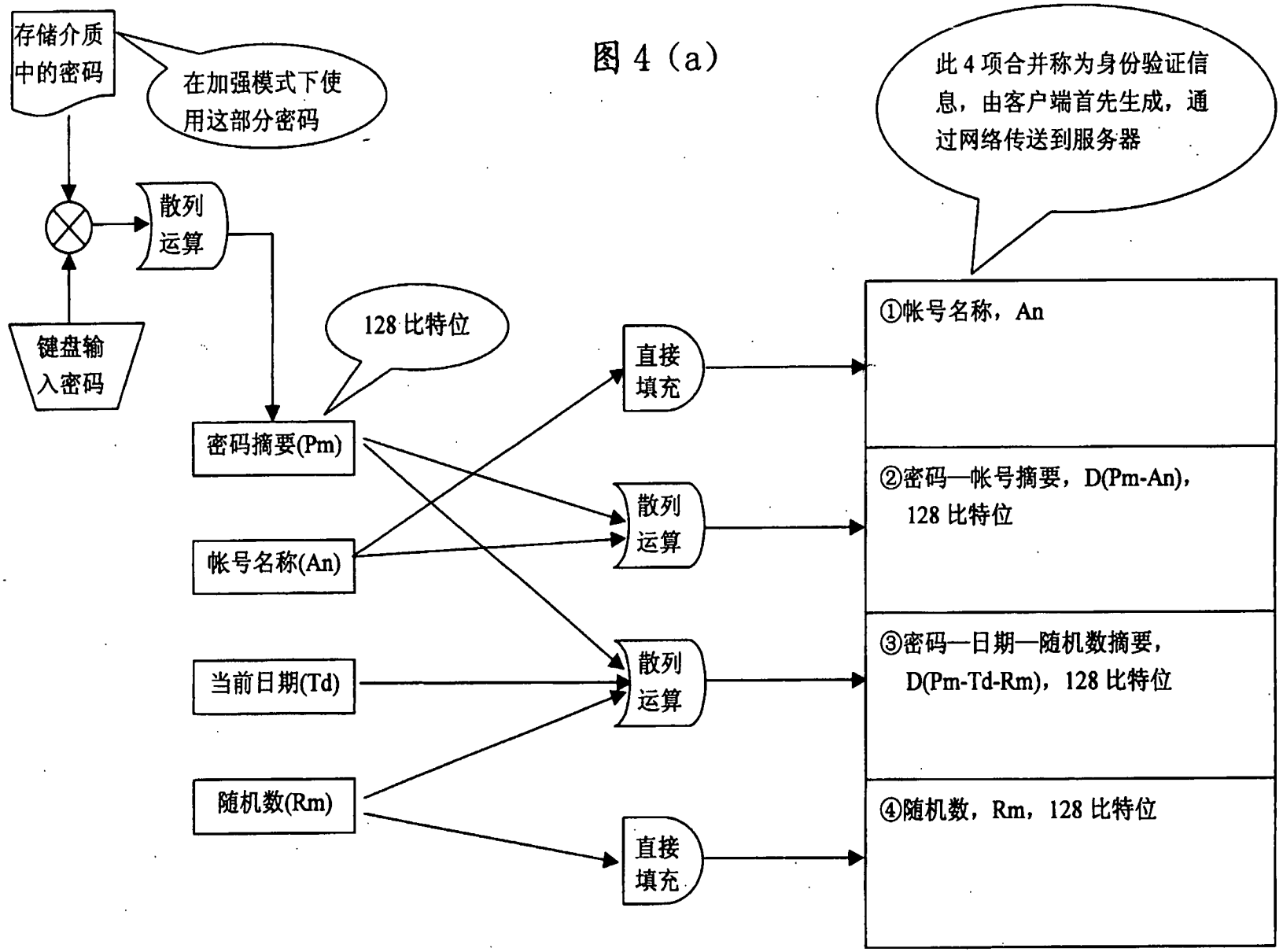

存储介质中的密码
在加强模式下使用这部分密码
散列运算
键盘输入密码
128 比特位
密码摘要(Pm)
帐号名称(An)
当前日期(Td)
随机数(Rm)
直接填充
散列运算
散列运算
直接填充
此 4 项合并称为身份验证信息，由客户端首先生成，通过网络传送到服务器
①帐号名称，An
②密码—帐号摘要，D(Pm-An)，128 比特位
③密码—日期—随机数摘要，D(Pm-Td-Rm)，128 比特位
④随机数，Rm，128 比特位

图 4（a）

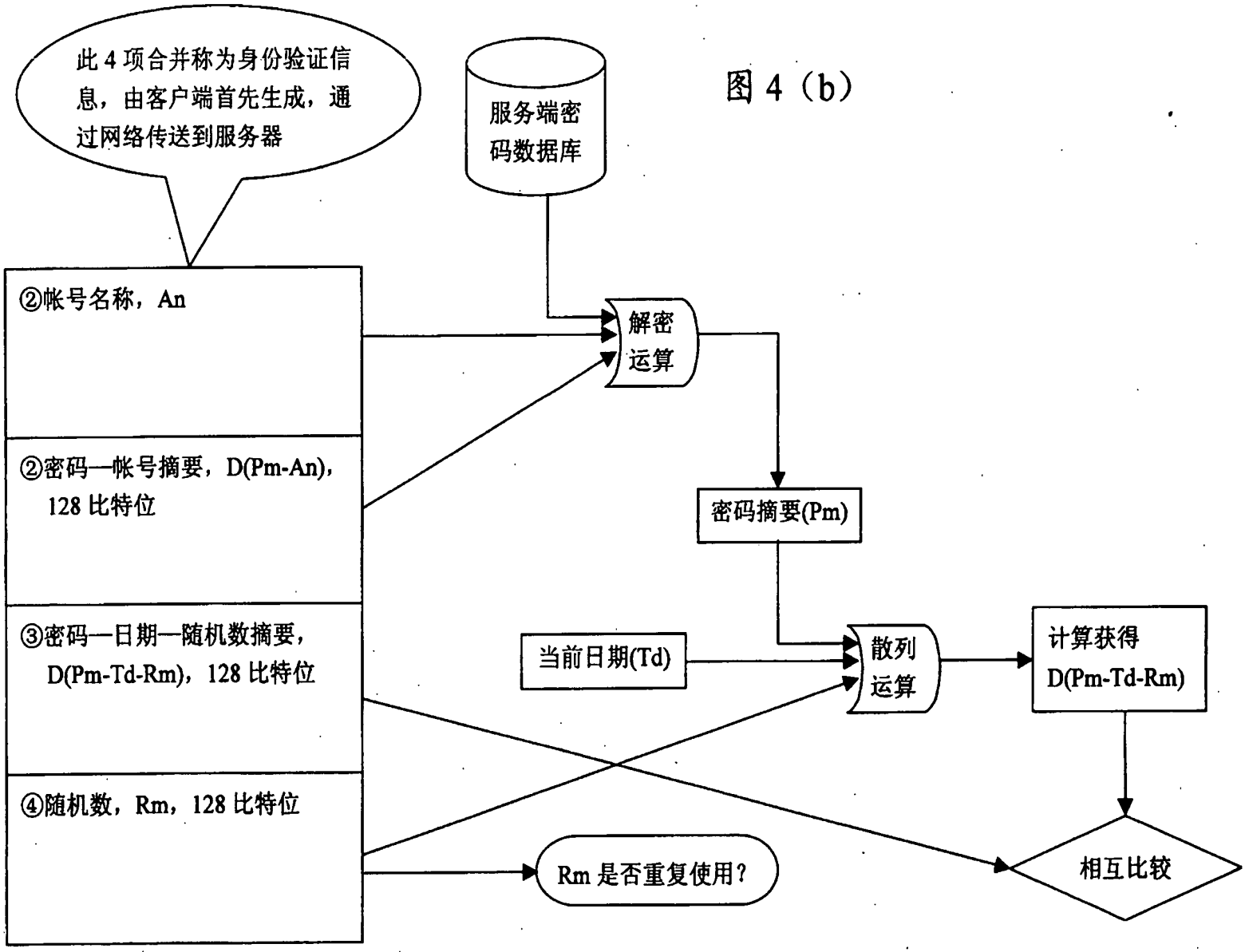
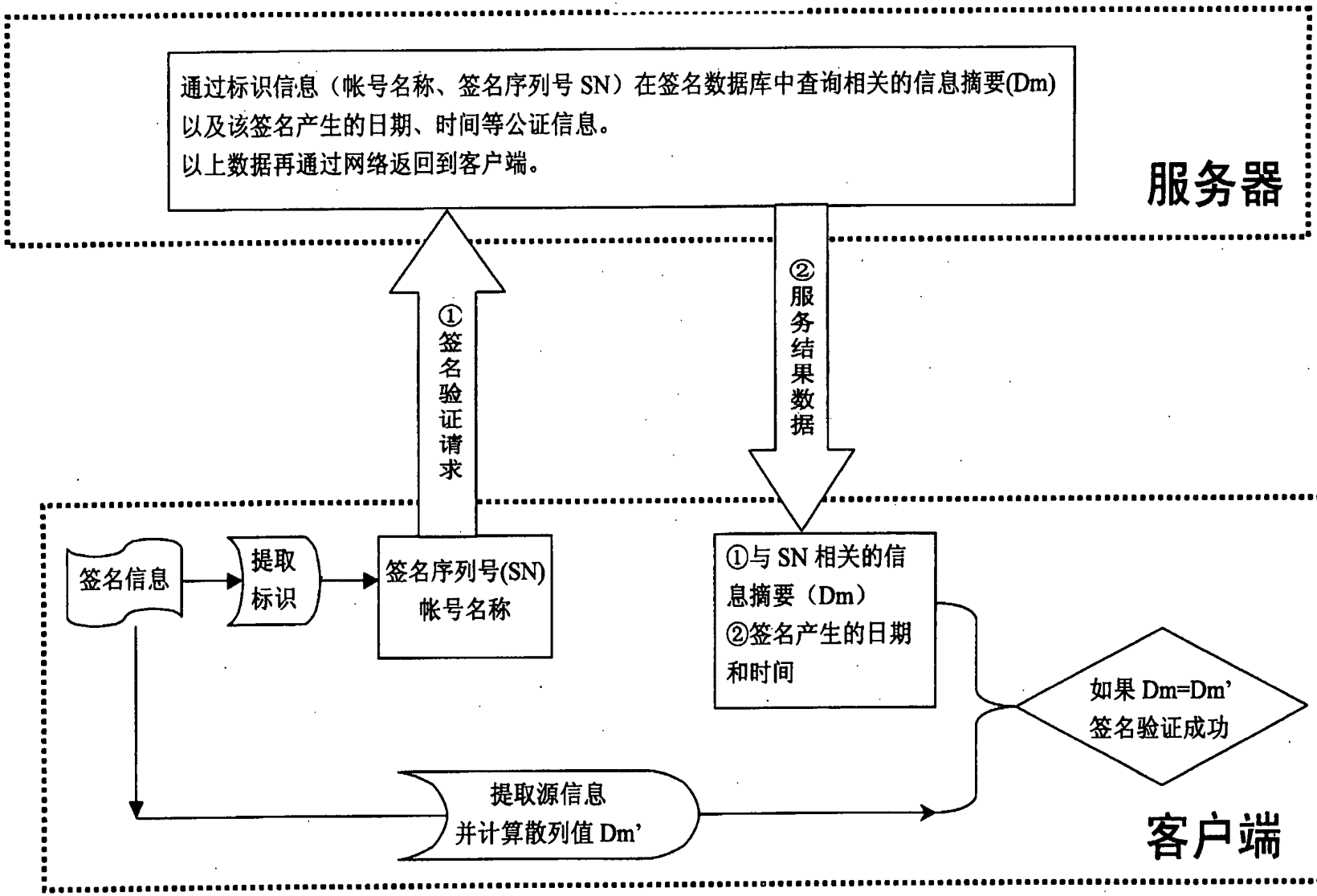

此 4 项合并称为身份验证信息，由客户端首先生成，通过网络传送到服务器
服务端密码数据库
②帐号名称，An
②密码—帐号摘要，D(Pm-An)，128 比特位
③密码—日期—随机数摘要，D(Pm-Td-Rm)，128 比特位
④随机数，Rm，128 比特位
解密运算
密码摘要(Pm)
当前日期(Td)
散列运算
计算获得 D(Pm-Td-Rm)
相互比较
Rm 是否重复使用？

图 4（b）

图 5


通过标识信息（帐号名称、签名序列号 SN）在签名数据库中查询相关的信息摘要(Dm)
以及该签名产生的日期、时间等公证信息。
以上数据再通过网络返回到客户端。
服务器
①签名验证请求
②服务结果数据
签名信息
提取标识
签名序列号(SN)
帐号名称
①与 SN 相关的信息摘要（Dm）
②签名产生的日期和时间
如果 Dm=Dm'
签名验证成功
提取源信息
并计算散列值 Dm'
客户端

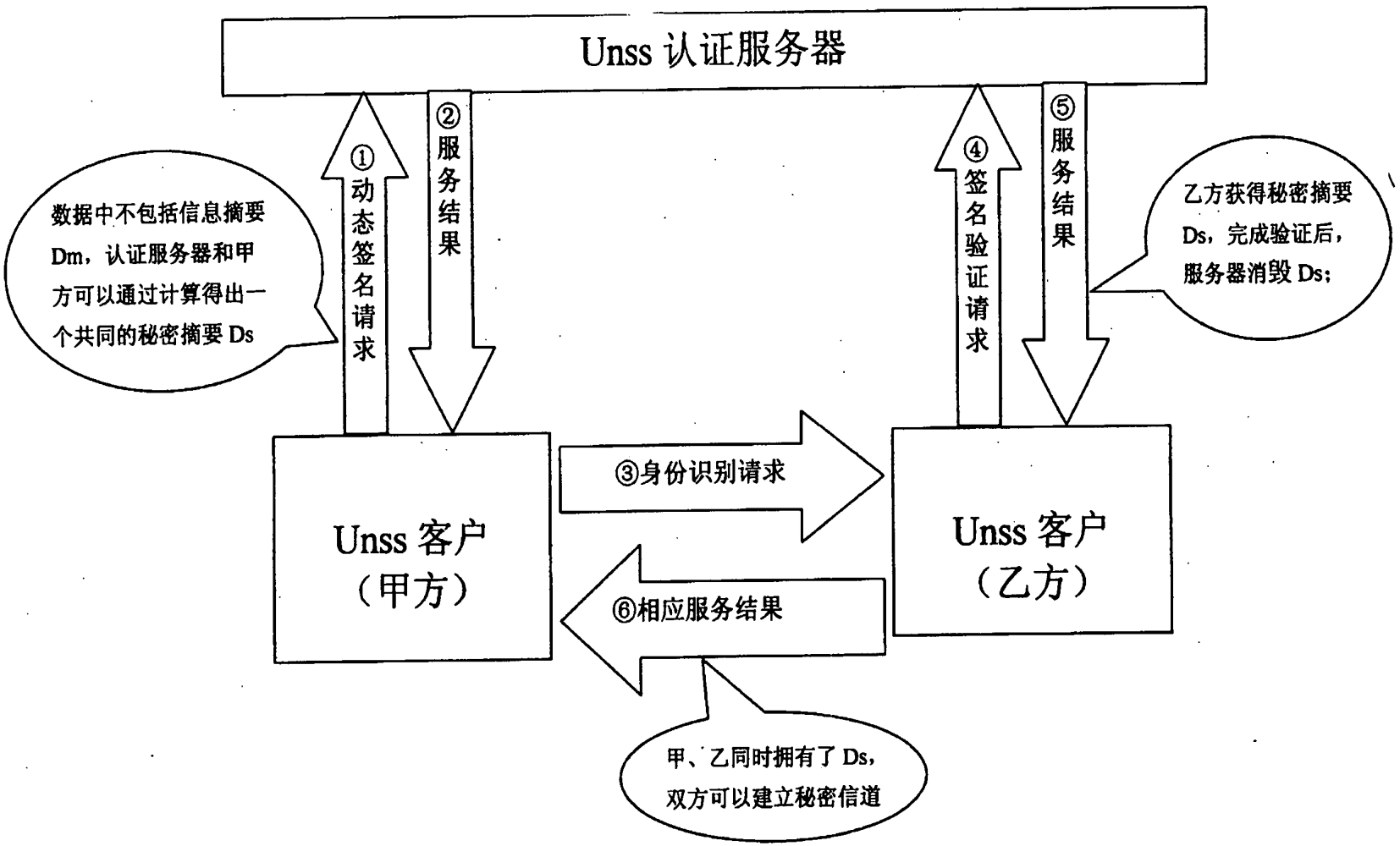

Unss 认证服务器
①动态签名请求
②服务结果
④签名验证请求
⑤服务结果
数据中不包括信息摘要Dm，认证服务器和甲方可以通过计算得出一个共同的秘密摘要 Ds
乙方获得秘密摘要Ds，完成验证后，服务器消毁 Ds；
③身份识别请求
Unss 客户（甲方）
Unss 客户（乙方）
⑥相应服务结果
甲、乙同时拥有了 Ds，双方可以建立秘密信道

图 6

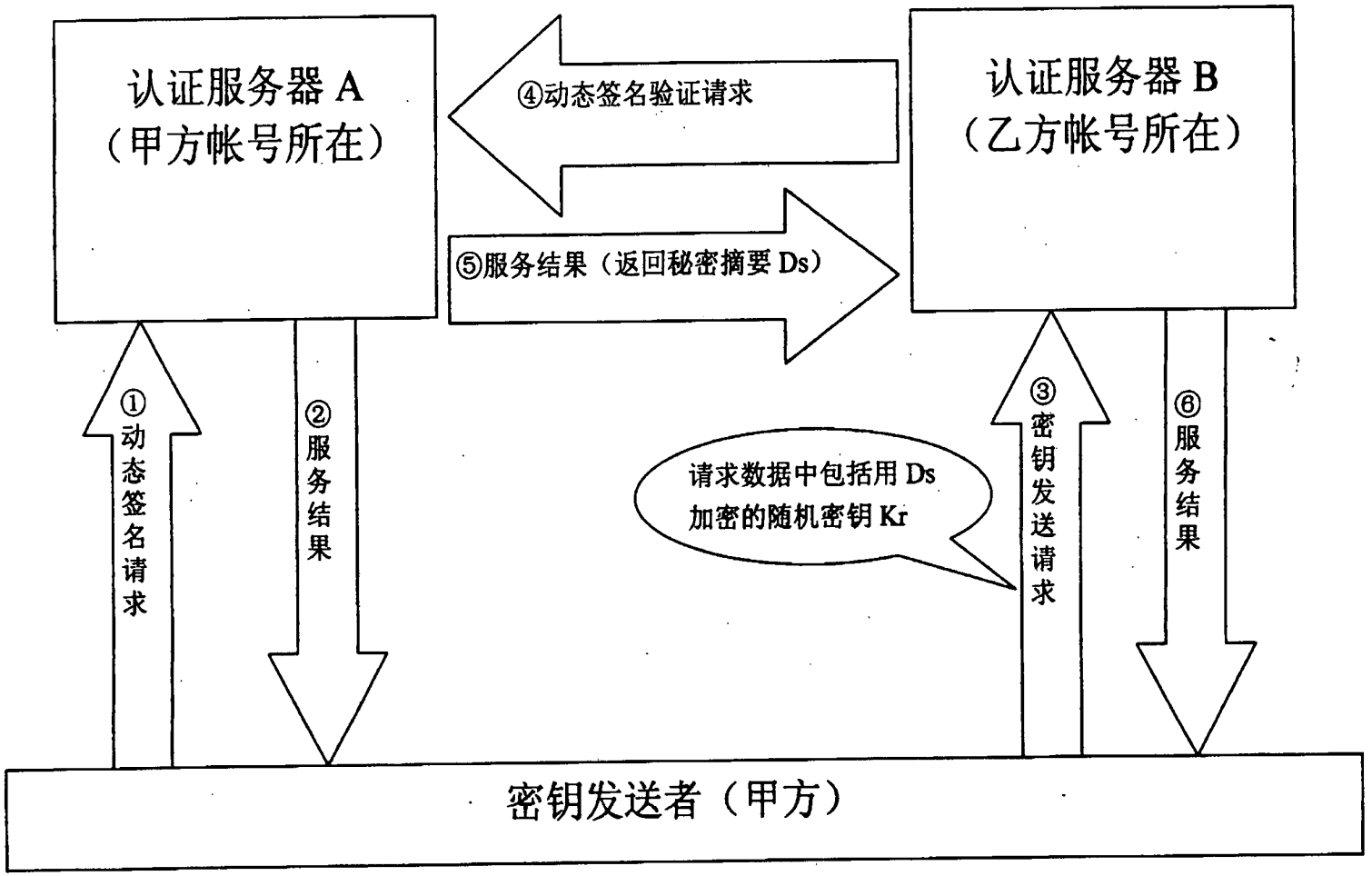

认证服务器 A
（甲方帐号所在）
认证服务器 B
（乙方帐号所在）
④动态签名验证请求
⑤服务结果（返回秘密摘要 Ds）
①动态签名请求
②服务结果
③密钥发送请求
⑥服务结果
请求数据中包括用 Ds
加密的随机密钥 Kr
密钥发送者（甲方）

图 7

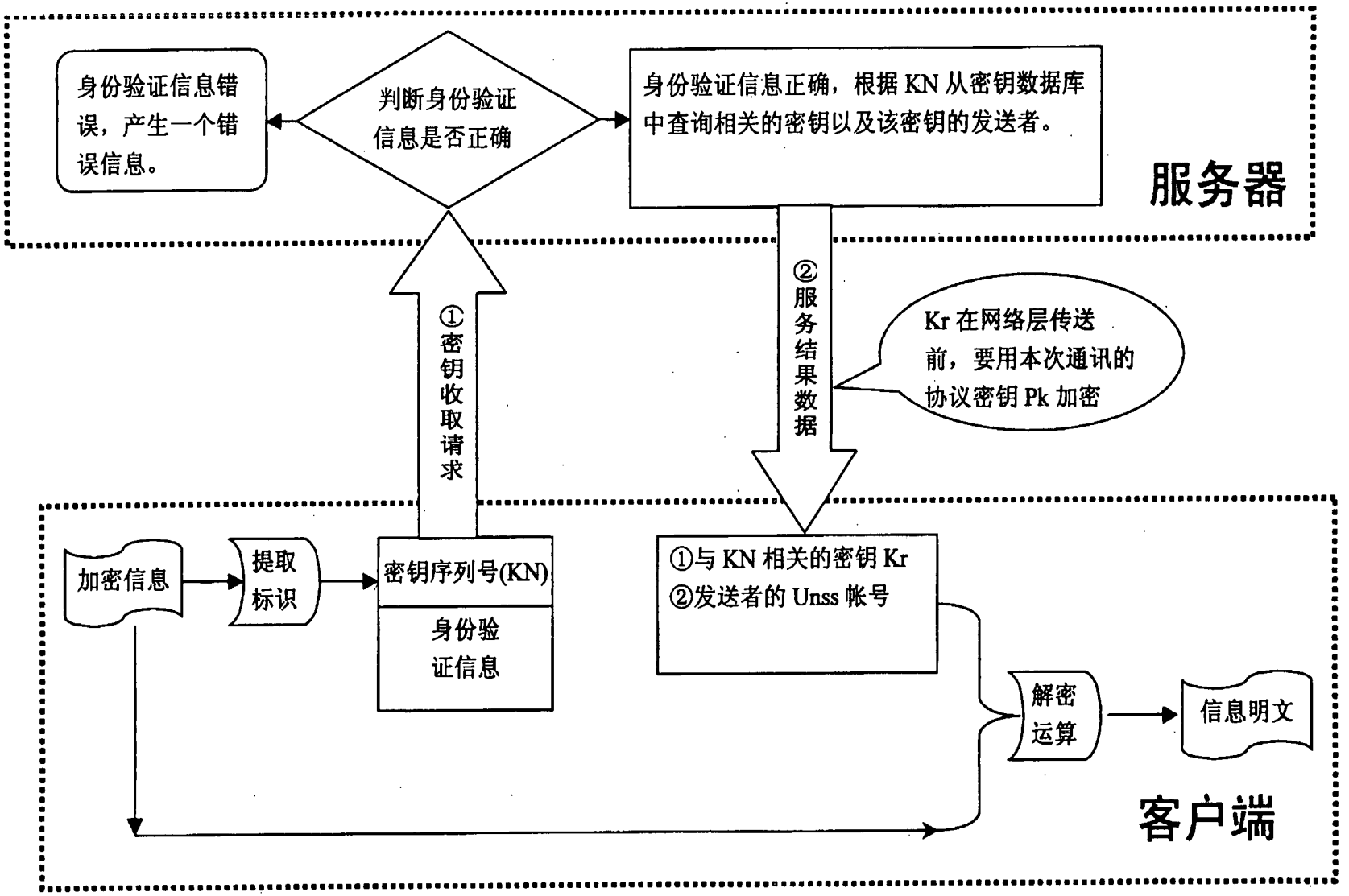

身份验证信息错误，产生一个错误信息。
判断身份验证信息是否正确
身份验证信息正确，根据 KN 从密钥数据库中查询相关的密钥以及该密钥的发送者。
服务器
①密钥收取请求
②服务结果数据
Kr 在网络层传送前，要用本次通讯的协议密钥 Pk 加密
加密信息
提取标识
密钥序列号(KN)
身份验证信息
①与 KN 相关的密钥 Kr
②发送者的 Unss 帐号
解密运算
信息明文
客户端

图 8

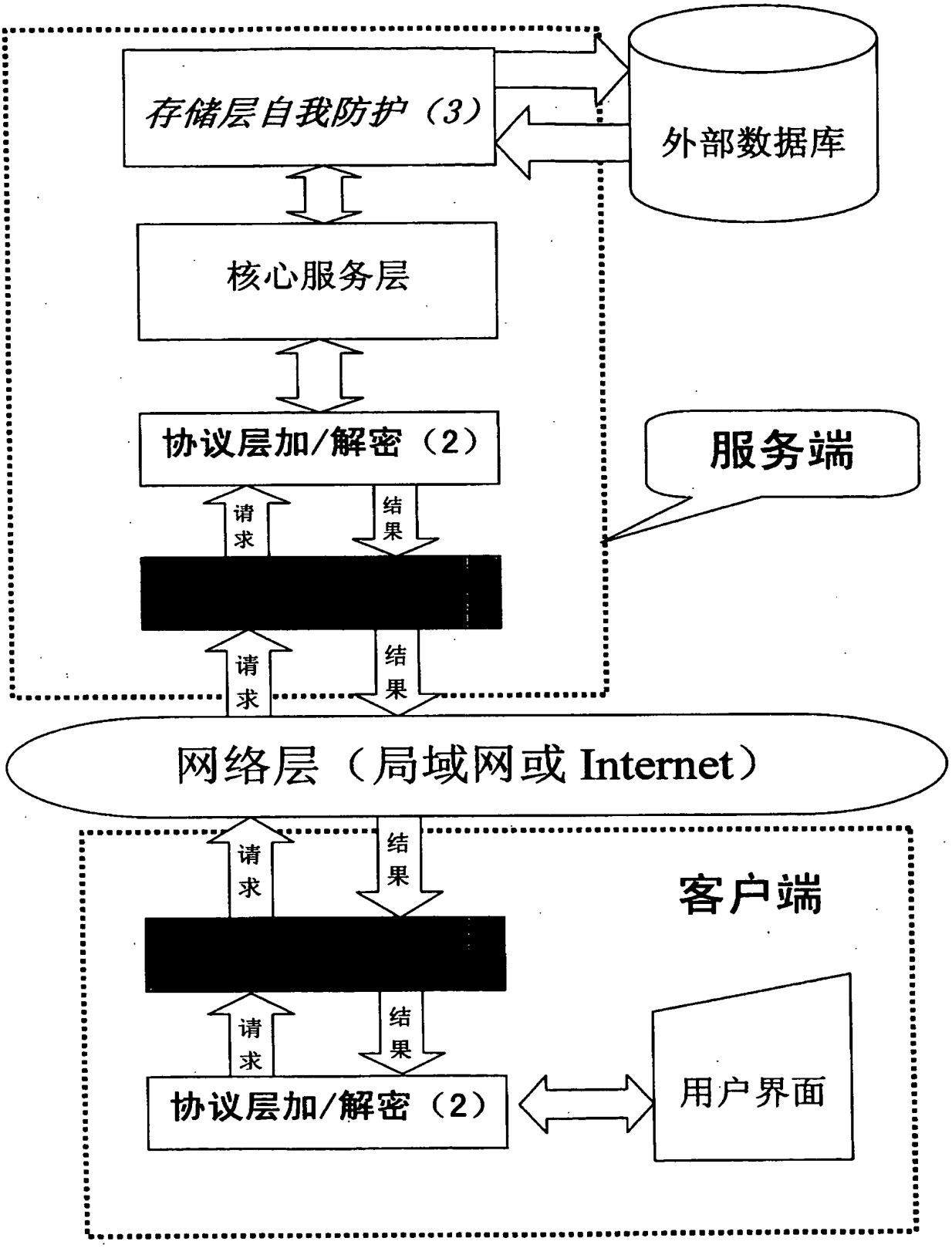

存储层自我防护（3）
外部数据库
核心服务层
协议层加/解密（2）
服务端
请求
结果
请求
结果
网络层（局域网或 Internet）
请求
结果
客户端
请求
结果
协议层加/解密（2）
用户界面

图 9

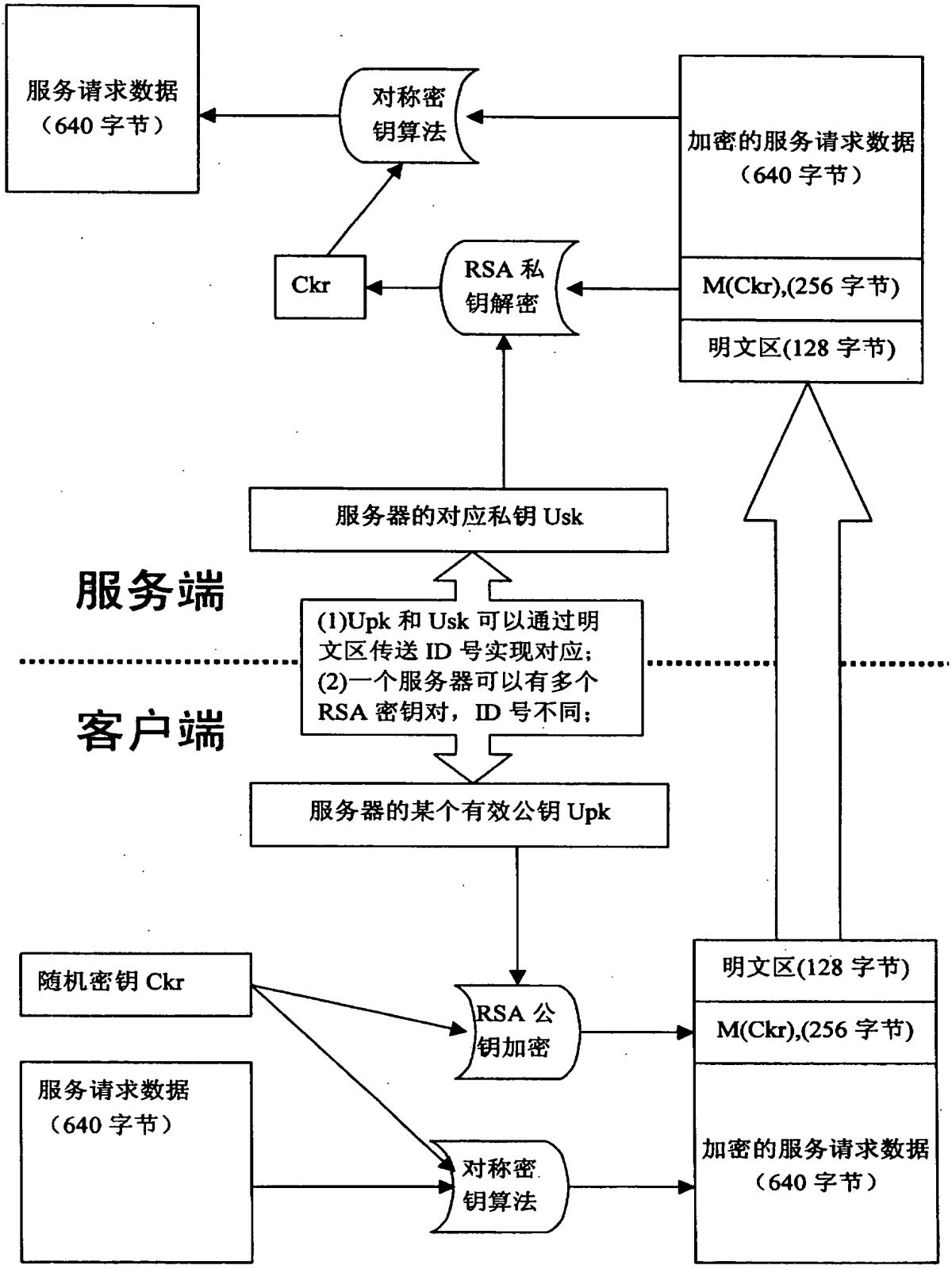

服务请求数据
（640 字节）
对称密
钥算法
加密的服务请求数据
（640 字节）
Ckr
RSA 私
钥解密
M(Ckr),(256 字节)
明文区(128 字节)
服务器的对应私钥 Usk
服务端
(1)Upk 和 Usk 可以通过明
文区传送 ID 号实现对应；
(2)一个服务器可以有多个
RSA 密钥对，ID 号不同；
客户端
服务器的某个有效公钥 Upk
随机密钥 Ckr
RSA 公
钥加密
明文区(128 字节)
M(Ckr),(256 字节)
服务请求数据
（640 字节）
对称密
钥算法
加密的服务请求数据
（640 字节）

图 10

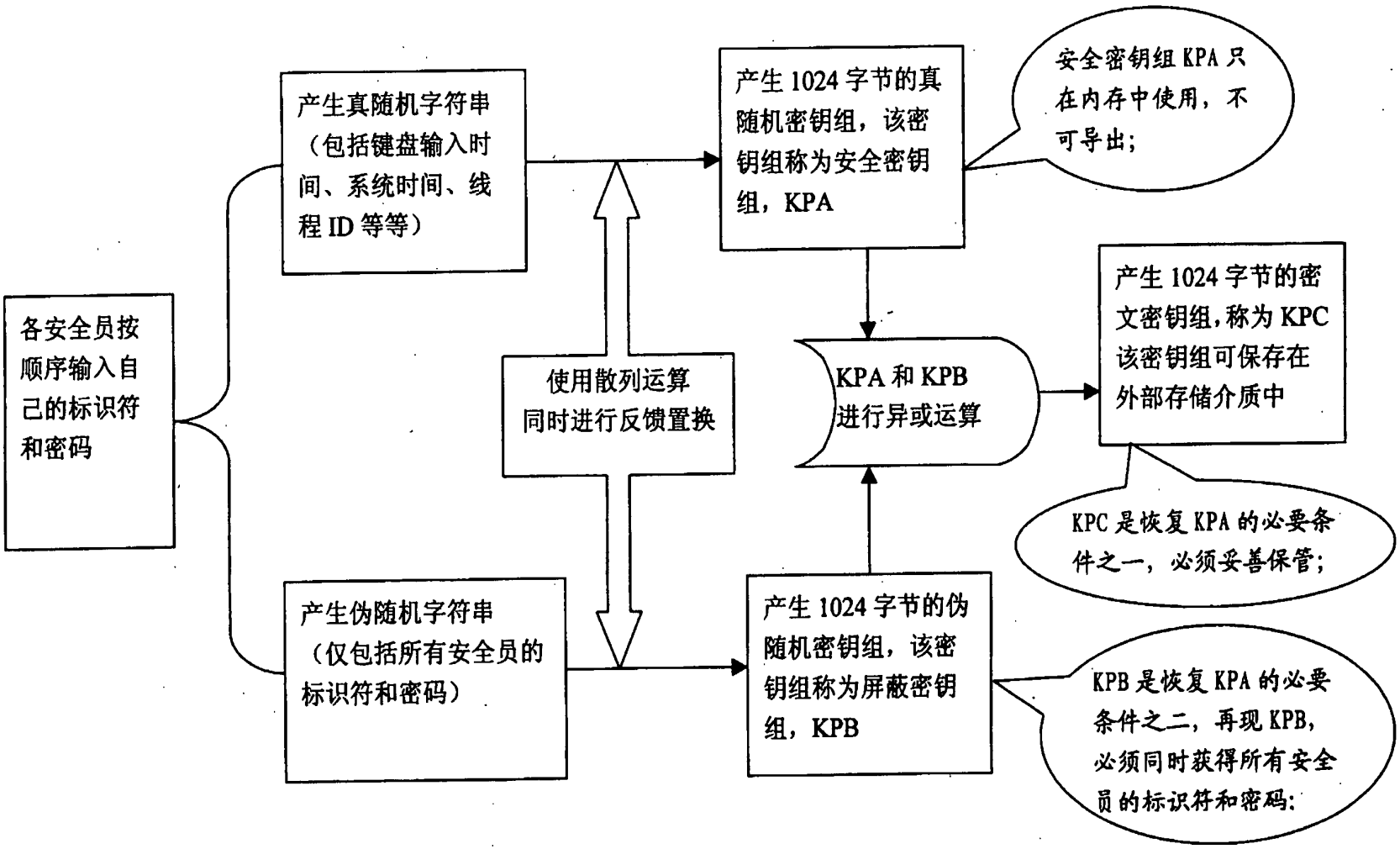

各安全员按顺序输入自己的标识符和密码
产生真随机字符串（包括键盘输入时间、系统时间、线程 ID 等等）
产生伪随机字符串（仅包括所有安全员的标识符和密码）
使用散列运算同时进行反馈置换
产生 1024 字节的真随机密钥组，该密钥组称为安全密钥组，KPA
安全密钥组 KPA 只在内存中使用，不可导出；
KPA 和 KPB 进行异或运算
产生 1024 字节的密文密钥组，称为 KPC 该密钥组可保存在外部存储介质中
KPC 是恢复 KPA 的必要条件之一，必须妥善保管；
产生 1024 字节的伪随机密钥组，该密钥组称为屏蔽密钥组，KPB
KPB 是恢复 KPA 的必要条件之二，再现 KPB，必须同时获得所有安全员的标识符和密码；

图 11

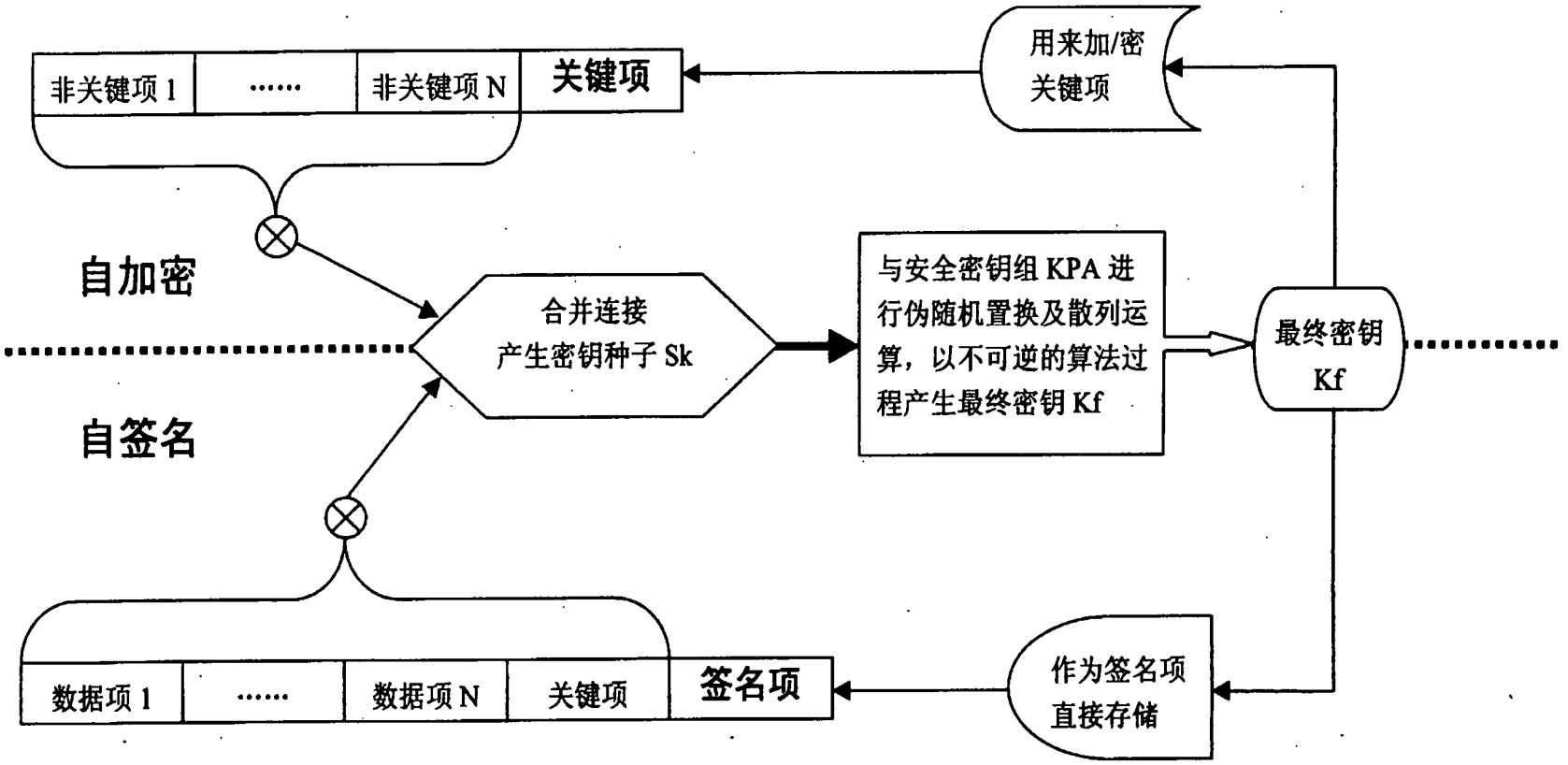

非关键项 1
……
非关键项 N
关键项
用来加/密
关键项
自加密
合并连接
产生密钥种子 Sk
与安全密钥组 KPA 进行伪随机置换及散列运算，以不可逆的算法过程产生最终密钥 Kf
最终密钥
Kf
自签名
数据项 1
……
数据项 N
关键项
签名项
作为签名项
直接存储

图 12

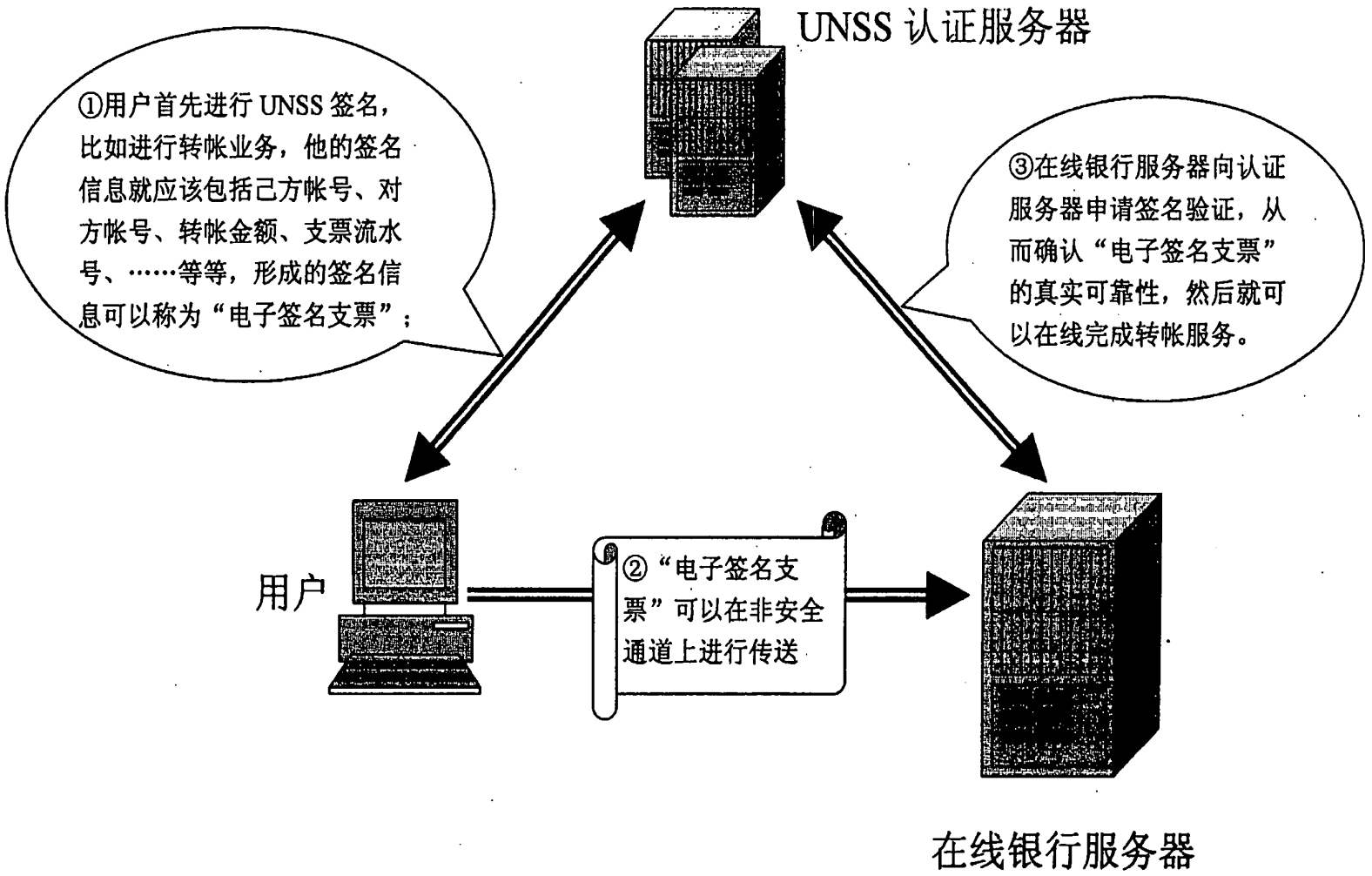

UNSS 认证服务器
①用户首先进行 UNSS 签名，比如进行转帐业务，他的签名信息就应该包括己方帐号、对方帐号、转帐金额、支票流水号、……等等，形成的签名信息可以称为“电子签名支票”；
③在线银行服务器向认证服务器申请签名验证，从而确认“电子签名支票”的真实可靠性，然后就可以在线完成转帐服务。
用户
②“电子签名支票”可以在非安全通道上进行传送
在线银行服务器

图 13